

Collaborative Computer-Aided Parameter Exploration for Music and Animation

Daryl H. Hepting and David Gerhard

Department of Computer Science
University of Regina
Regina, Saskatchewan
S4S 0A2 CANADA
{hepting, gerhard}@cs.uregina.ca

Abstract. Although many artists have worked to create associations between music and animation, this has traditionally be done by developing one to suit the pre-existing other, as in visualization or sonification. The approach we employ in this work is to enable the simultaneous development of both music and sound from a common and rather generic central parameter variation, which may simply indicate a structure for periodic repetitions. This central parameter variation is then simultaneously mapped to appropriate musical and graphical variables by the musician and the animator, thereby contributing their own interpretations. The result of this mapping is then rendered in an intermediate form where music and animation are allowed to iteratively influence each other. The main piece of software in this development is the system which allows exploration of parameter mappings. The software interface allows both musician and animator to meaningfully experiment with the other's mappings since the interface permits access in a common form, without requiring additional skills to interpret.

1 Introduction

Throughout recent history, various artists have explored the relationship between music and colour and between a soundtrack and its animation. While early attempts to generate both sound and music from the same composition met only with mild success, it has been observed by artists like Barry Spinello [?] that “we are now at the point where film and music have gone their separate ways so that the only conciliation of the two seems to be some form of ‘synchronization’; that is music will be composed for an existing film sequence or vice versa.”

Current examples of this sort of conciliation are the work done by Lytle at *Animusic* [?], where animation is generated to correspond to a pre-existing MIDI score, and the work done by a composer creating the score for a movie once that movie is complete. Popular music visualizers, like WinAmp [?] and iTunes [?] use a different approach that is reactive in nature since they are intended to create a visual for the sound being heard at the moment. This reaction is not dissimilar to the way pianists may have approached the task of accompanying silent films in

years past, although repetition for them might have improved their interpretation for subsequent screenings. While Lytle’s work creates visualizations for sounds at the note level, WinAmp makes visualizations at the sample level. However, neither allows for any interaction between the sound and the visualization.

Examples abound of either the music or the animation driving the development of the other, either in realtime or not. However, little has been done to explore a process that allows the music and animation to be developed together by mapping each from a single central parameter variation. The resulting intermediate music and animation can further be used to inform each other, to create more interconnections and nuances in the final work. The principle advantage of this parameterization is that musician and animator can both interact with the piece using the same parameter set so that one can affect the portion under the domain of the other. This empowers the animator to say “music like this” or the musician to similarly say “pictures like this.”

This paper describes a process that allows musicians and animators to work together and collaborate closely by virtue of computer-support for the parameterization and exploration of the resultant parameter spaces. The rest of the paper is organized as follows. Section 2 describes the motivation and prior work. Section 3 describes the present approach and the use of computer-based tools to support it. Section 4 presents an example of how this approach can be used in the development of an integrated work. Section 5 discusses conclusions from our experience and directions for future work.

2 Background

There are different ways to combine music and animation and there are different ways to conceive of the relationship between the two. Schillinger [?] and Whitney [?], to name two, had their own precise ideas about this relationship, which Moritz [?] distinguished as quantitative and qualitative. Bute and Schillinger had an long-lasting collaboration to explore Schillinger’s system of music through Bute’s animation. The of the previous work can be divided into *visualization* of audio and *sonification* of video.

2.1 Visualization

Audio visualization has been a popular application of cross-modal media generation since computers made it possible to analyze sound in real time. In general, there are two types of audio visualizers: waveform-driven and event-driven. Waveform-driven visualizations encompass the visual plug-ins available in media players such as WinAmp and iTunes. The sound waveform (and often a rudimentary frequency analysis of said waveform) is made available to a plug-in program which renders visual data based on the waveform data. Usually, this consists of drawing and manipulating the actual waveform or spectrum, in the style of an oscilloscope or spectrum analyser. and adding colours and transitions that are

unrelated to the sound. This method is consistently successful in generating visual content related to the sound, however it can only react—the visual medium depends entirely on the acoustic waveform in the current time frame, or more commonly of an immediately past time frame. This method can be extended to predict future events, but little work has been done on this, primarily because the iTunes and WinAmp visualizations are doing precisely what they were designed to do: add some visual effects to what is primarily an audio experience. These visualizations are incapable of producing anything contextually relevant, for example a picture of the musician. Their role is strictly augmentative. Even if it were part of the goal to produce a contextually *relevant* visualization, it is very difficult or even impossible to choose the extra visualization information (colour, movement etc) based on acoustic constructs because the available acoustic information is strictly limited to instant realtime waveform and spectrum information.

It should be noted that waveform driven realtime visualizations like iTunes and WinAmp can be augmented with contextual information from ID3 tags, which contain artist, album and year information, as well as *genre* information. Visualizer components can be developed to make use of this information, using specific colour palette for specific genres of music, however this data is limited to the accuracy of the person who entered these tags, and is a single characteristic, constant for the entire sound piece. If a piece of music changes from minor key to major key half way through, this change is not reflected in the ID3 tags, nor is it possible to reflect any structural information like this in the visualization.

The second class of audio visualization is one step removed from the actual waveform, to the *events* that cause the waveform. In this case, a set of parameters (the list of events) is used to generate both the video and audio media. The classic example of this is Wayne Lytle's Animusic pieces [?,?]. In this work, a MIDI (musical instrument digital interface) file is used to generate music, and an animation is also generated which corresponds to the music. Virtual musical instruments are designed which actuate according to the MIDI file, so that it appears that the virtual musical instruments are “playing” the music.

The Animusic work is very cohesive because it is animating the instruments or the likely sources of the acoustic events. It works because the domain is auditory and visual at the same time, in the same way that a moving mouth on a face tends to fuse with an auditory speech waveform into an apparant single entity.

Event-based visualizers like Animusic are inherently non-realtime because in many cases, visual events must *precede* the corresponding acoustic events. In order for a visual drum hit to appear to synchronize, or fuse, with the corresponding acoustic event, the drum stick must begin the motion before the event, so that the stick strike will occur at the same time as the sound. Another limitation with this technique is that visualizations can correspond only to acoustic events. An acoustic event without an apparent visual analogue, such as background ambience or continuo, must be visualized in an abstract way and this does not always fuse as well as directly realizable instruments like drums. This

abstract visualization, combined with the exaggeration of non-moving instruments like horns, may create some dissonance in the synchronization.

The fundamental limitation of this technique, as well as the realtime visualization mentioned above, in the context of this paper, is that both are inherently unidirectional. All relevant generated visuals are based entirely on the pre-recorded sound. In the best case, the realtime visualization technique can be applied to realtime sound input, and perhaps a person could play an instrument, watch the corresponding visualization, and change the way they play based on the visuals immediately past. Even in this case, all visuals are generated based entirely on the sound waveform, and as such, no *interaction* is possible.

2.2 Sonification

Sonification is a complementary technique, where sound is generated based on numerical or visual parameters [?,?]. Historically, sonification (also called auralization) has referred to the production of sound based on some data-driven parameters, as an augmentation to data visualization techniques. Sonification of visual representations has been done, but in a very formalized way. A straightforward historical example is the musical score: commercial systems exist which will scan a piece of sheet music and translate the resulting image into a musical representation such as MIDI or MusicXML [?], for playback or manipulation in a music editing program. A more general application of the sonification of graphical objects is composition by time-frequency diagrams, where a composer “draws” the score on a piece of paper in a time-frequency representation, and the computer scans the sheet and produces sound corresponding to the lines drawn by the composer. This is much more versatile than simple score scanning, since dynamic time-frequency constructs such as sweeps and noise clouds are available to the composer, however this method is not realtime and requires a development–evaluation cycle where a figure is drawn first, and after the figure is complete the new composition can be heard.

2.3 Combined Visualization and Sonification

Cross-modal systems have been proposed in the past which attempt to integrate visual and acoustic events into a contiguous whole. Of particular note is the work of Hahn et al. [?,?], where visual and acoustic events are generated together. Their approach seeks to synchronize the development of sound and *motion* by connecting the sound and motion parameters before rendering. Their work is primarily motivated by the addition of sound effects to graphic objects, such as wind-chimes or a spinning disk. The animation of the movement of the object is described using a set of parameters, and these parameters are used to synchronize and generate the associated sound.

In many ways, this is the inverse of Lytle’s event-driven music visualization described above. In this case, *visual* events are described in a parameter space and the corresponding *acoustic* objects are generated from this parameter space. The same limitations apply to this method as did to Lytle’s *Animusic*: the system

is inherently unidirectional and the sounds match the motion only inasmuch as the motion can be seen to generate the sound. Motions with no natural associated sounds cannot be linked to a sound by this method in a unified way.

This should not be taken as a criticism of these methods—*Animusic* and Hahn’s integrated approach are both exceptional pieces of work, for their specified domains. The automatic generation of sound effects (also called digital foley) is an interesting and difficult research domain in its own right. Rather, we see to show how our system of interaction is different from these previous systems. It is bidirectional rather than unidirectional, and iterative rather than single-pass. Each medium has the potential to be equally important in the final generation, and what is perhaps most important, the structure of the proposed system is not specific to the media being generated. The system could be expanded to any number of related media, all generated from the central abstract parameter set and augmented by interactions with the other media.

Although the translation of data from one medium to another is a well-established research area, in the above examples, and indeed in much of the related work, one medium has a strictly defined format from which parameters are extracted to generate the second medium. Little has been done in the way of the simultaneous, bidirectional generation of visual and acoustic objects from a common set of parameters. The next section describes a computer-supported approach to facilitate such simultaneous generation.

3 Approach

Parameterizations of music and visual art are available in a variety of forms. Such abstractions are useful to enable a richer exploration of the subject material by broadening one’s concept of the material. For example, Stockhausen’s music explored different parameters [?]. Boden [?] discusses conceptual spaces defined by stylistic conventions, and how to expand those spaces with three rules: “drop a constraint”, “consider the negative”, and “vary the variable.” The card deck, known as *Oblique Strategies* [?] also attempted to provide a means to achieve such an expansion.

However, the conceptual spaces might not need so much expansion as more effective exploration. In discussing his work for the film *Arabesque*, Whitney says (in 1980) that “this program concept was not deeply ‘mined’ for its fullest possibilities. ...It is significant to realize how meager my experience was in exploring the *Arabesque* material, and how limited all exploration is to date, in this area of computer graphics.”

Whitney used a *manual* approach, according to Kochhar et al. [?] with respect to the computer. While the computer provided the raw capabilities, Whitney was responsible for putting everything together. Other relationships between human and computer identified by Kochhar et al. [?] are *automatic*, at the other end of the spectrum from manual, and *augmented* which seek to balance manual and automatic characteristics to support the user in his or her task. Automatic systems, which generate solutions with a minimum of user input, are precluded

from this task because, as Whitney said “art is a matter of ‘judgment – not calculation.’ ” In terms of visual decision making, Bertin’s process [?] comprised matrix analysis of the problem (questions are defined); graphic information-processing (answers are discovered); and graphic communication (answers are communicated). For him, it was clear that this work could never be automated because no machine would ever be able to solve this problem of imagination.

Between these extremes are augmented systems that enable computers and humans to work together. In a “good” interface, according to Baecker et al. [?] human and computer “should augment each other to produce a system that is greater than the sum of its parts. Computer and human share responsibilities, each performing the parts of the task that best suit its capabilities. The computer enhances our cognitive and perceptual strengths, and counteracts our weaknesses. People are responsible for the things the machine cannot or should not do.” For Baecker et al, [?], human capabilities include judgment and creativity and computer capabilities include patience and reliable memory.

The *cogito* system [?,?] has been developed as a means to preserve the user’s ability to exert judgment while exploring a large, complex parameter space. Points in the space are constructed from all possible combinations of values from the different parameters. For example, given 3 parameters each with 10 possible values, the space has $10^3 = 1000$ points. Foley and Ribarsky [?] wrote that only automatic methods could be used effectively with large parameter spaces because otherwise the users would be overwhelmed. Users of the *cogito* system have not been overwhelmed, however, even though they deal with millions of possible combinations [?]. The system, depicted schematically in Figure 1, allows users to apply their own judgment when assessing alternatives automatically generated by the system, which has the patience to realize these different combinations and keep track of them.

The system functions in two ways: first it allows people to explore different relationships among parameters. By reorganizing the view of the parameter space, as shown in Figure 2, it is possible to transform one’s perception of the parameter space from Klondike (difficult to navigate) to homing (easy to navigate) and find a desired solution [?]. Secondly, it records and manages the exploration of space, so that the space can be explored in a systematic way and one can return to previous explorations to examine other options at decision points. In this way, the system is an example of a numerical experimentation environment [?]. Where there are examples of interest, the user may click to select them. The subsequent space of alternatives contains all possibilities consistent with those selections. For each parameter, the list of acceptable values comes from those in the selected exemplars.

The *Design Galleries* concept [?] has some interesting features - namely that it can show alternatives that are good examples of a set of values, however the evaluation function has to be specified *a priori*. The *cogito* system permits results to be partitioned based on qualitative or quantitative differences in values within one or more parameters.

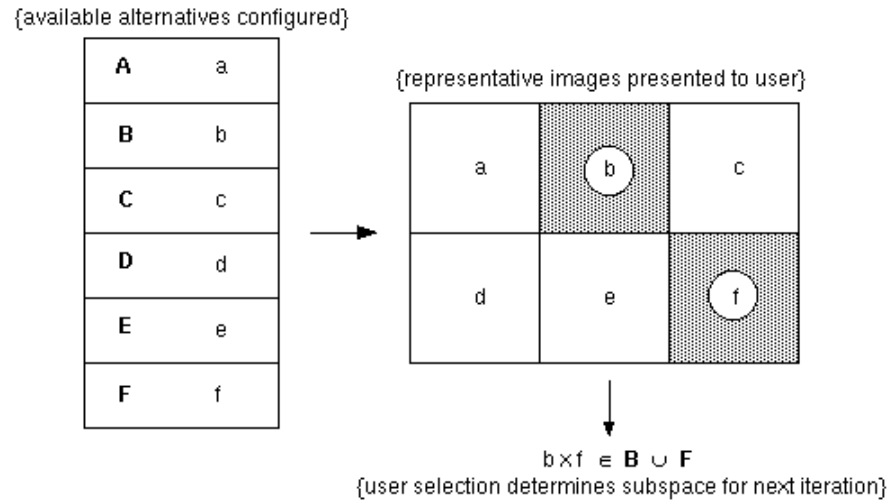


Fig. 1. Schematic look at the hierarchical interface: the space of available alternatives is grouped according to user-specified criteria. Each group (A – F) has a representative element (a – f) which is displayed to the user. The subspace for the next search iteration is based on the user selection (b and f).

The process of generating media from a parameter space is illustrated in Figure 3. It begins with the selection of a central parameter variation, followed by the selection of a mapping strategy to move from the central parameters to the media. A separate, independent mapping strategy can be selected for each medium in question, in this case for sound and video. Once the mapping strategy has been selected, the media are generated. At this stage, there are three options. The generated media can be taken as complete, the central parameter variations can be re-visited, or a set of *intermediate* parameters can be derived from the generated media. These extracted parameters are then mapped to media-specific constructs, and the media is re-rendered. Each parameter set (central and intermediate) can be re-visited as many times as necessary.

This parameter variation is expressed in generic terms so that both musician and animator have equal access to its modification. Such variations could be constructed in different ways: mathematically in terms of trigonometric functions or visually by sketching curves then copying, scaling, and translating them. Figure 4 illustrates one set of parameters used by Whitney [?]. It shows 12 points and the differential movement applied to each so that at the end of the cycle, the points are once again coincident. This illustrates in general Whitney's ideas about digital harmony, which he described in the context "the relationship of the three terms: **differential**, **resonance**, and **harmony**. First, motion becomes patterns if objects move *differentially*. Second, a resolution to order in patterns of motion occurs at points of *resonance*. And third, this resolution at resonant

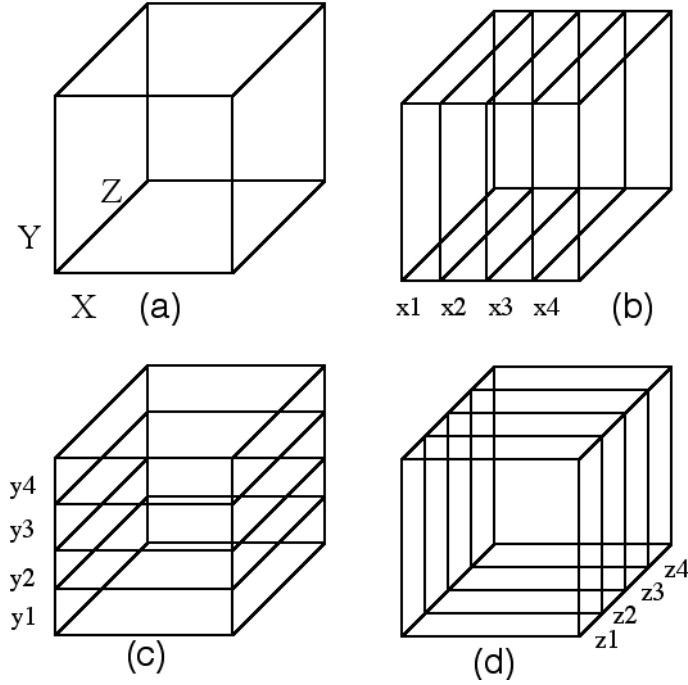


Fig. 2. Consider a three-dimensional space, depicted in the top left, with axes X , Y , and Z . Organizing the space in terms of any of those 3 axes leads to the other states depicted. If elements in component X are chosen sequentially, those in Y and Z can be selected randomly to give a sense available options.

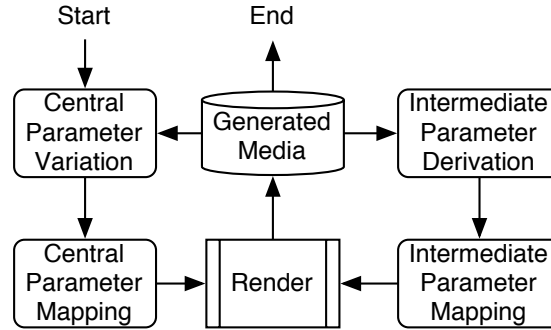


Fig. 3. The process of generating media from a parameter space. Begin by choosing a central parameter variation which can be mapped separately to parameters in the music and animation. Once the media has been generated, intermediate parameters can be extracted and mapped, and the central parameter set can be re-mapped until a desired effect is achieved. At each iteration, the generated media is available.

events, especially whole number ratios, characterizes the differential resonant phenomena of visual *harmony*.”

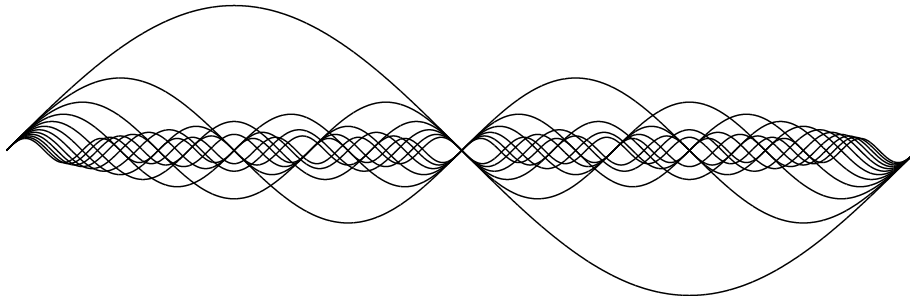


Fig. 4. Parameter variations illustrating Whitney’s concept of digital harmony (after Whitney).

This central parameter variation provides what can be considered the *score* of the integrated piece being composed. As such, the central parameter variation is represented with each parameter on a separate line, as in Figure 5. The score analogy is appropriate for musicians but also for animators since it is very similar to features present in most non-linear video editing programs and multimedia authoring systems. Such a representation enables direct access to any point in the work, allowing it to be played. The common and unbiased form of representation helps both musicians and animators contribute to the integrated work, without requiring a great deal of effort on the interface which has a low syntactic burden and seeks to minimize the gulfs of execution and evaluation [?].

When an orchestra plays a piece of music, a score is used to tell each instrument what to do. In the same way, the central parameters can be considered a score for the generation of the sound and animation. Each acoustic and visual event is generated based on the way the parameters change. The first level of media creation, then, can be considered as an orchestra playing a score. Once the first pass of media creation is complete, parameters are extracted from each medium and used to augment the corresponding medium, so extracted visual information (like the area of the figure) is used to augment the audio signal, while extracted audio parameters (like the loudness of the waveform) are used to augment the visual signal. These new parameter tracks are added to the overall score to generate the next level of audio and visual media. This process is closer to the idea of a group of improvisational jazz musicians playing a song together. Each musician has the lead sheet in front of them so that they are all playing the same song, but each individual then can alter or augment their playing based on the interaction with the rest of the musicians. The lead sheet (the score, the central parameters) is fixed, and when used as a starting point, ensures cohesion between the players, but it is not the whole story, and the final music generated by the players is more than the sum of the individual interactions.

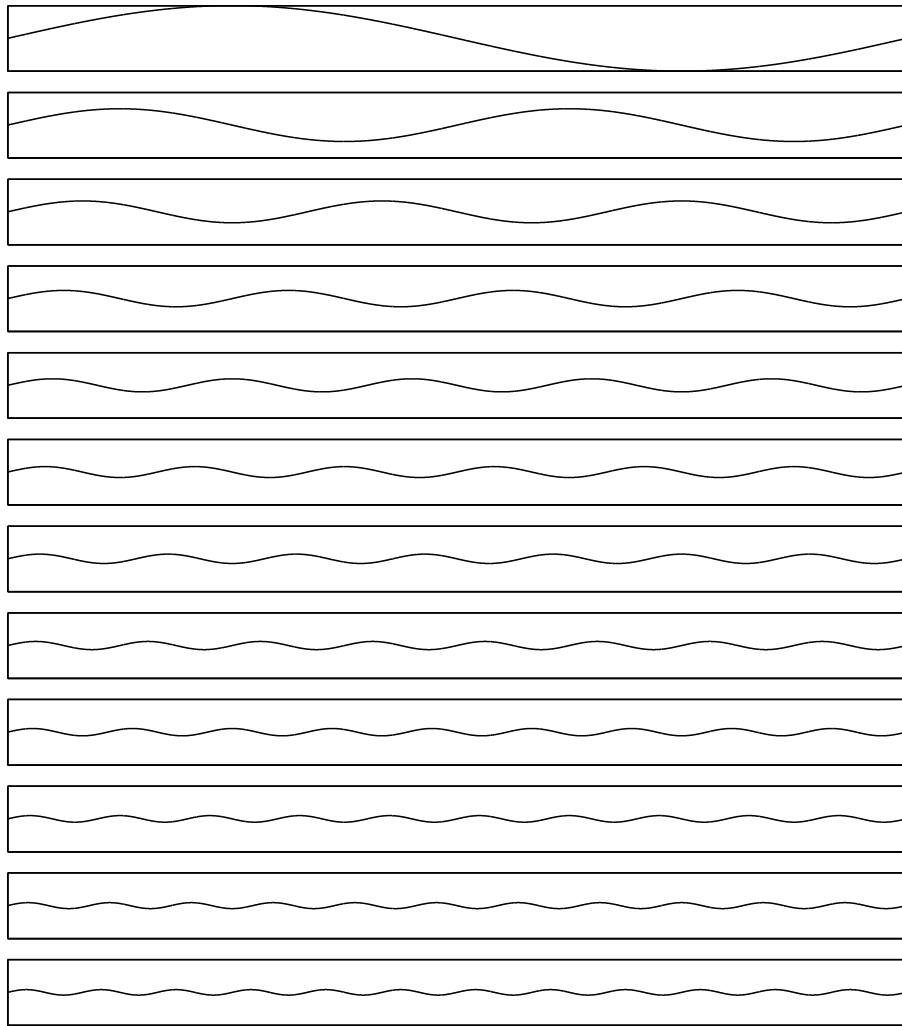


Fig. 5. The Whitney parameter variations from Figure 4 , written in our parameter score format.

The next phase involves the musician and animator each creating mappings from the central parameter variation to their respective media. It could be done in a variety of ways: algorithmically by modulating some facet of the media as a function of a particular central parameter component, or by choosing one or more keyframes and fitting them onto the central parameter variations. These exemplars may come from a variety of sources. The musician may play sounds or phrases on a keyboard, and the artist might draw keyframes to be analysed. The *cogito* system can be used to present different key visual frames or sounds for evaluation by the user.

Once the mapping has been established, the work can be rendered in a preliminary fashion. At this point, additional parameters can be derived from both the music and the animation. These parameters can be used to influence and add depth to the work in its present preliminary form. These additional parameters can be used either to map continuously onto an effect (such as adding reverb to a sound) or to drive a threshold function which can trigger an associated event (such as adding a musical note event). The *cogito* system could again be used to explore the different applications of the derived parameters within the whole work, serving as the basis for further iteration.

The goal of the *cogito* system is to allow the exploration of a space to find the particular combinations that meet the needs of the user. Simon [?] called these *satisficing* solutions. We get what is desired at a specific point in time, recognizing that what is desired may change over time. It is helpful then to be able to go back and revisit past decisions. In moving towards a piece that is accepted by a wide audience, we may apply constraints from a range of sources, to improve the final product.

4 Example

A short experimental piece, entitled *Triangularhythmic*, was created by the authors with the help of Matthew McKague. The central parameter variation used is based on groups of three. This was chosen with foreknowledge of the visual space that would be explored, and with an idea of the animation mapping that might be used. The visual space is that of fractals [?] in general, and in particular those generated from sets of contractive affine transformations [?]. The sonic space is generated using rhythmic loops designed to fit the fractal transformations at the central parameter level, and composed using Soundtrack [?].

The parameter score for this example appears in Figure 6. The top five parameter tracks correspond to central parameters used to generate the first pass acoustic and visual media. The bottom two parameter tracks are examples of extracted parameters. Figure 7 shows the Fourier spectrum of the generated soundtrack.

There are several benefits of this representation. One can see the whole timeline at once, or isolate a specific segment, and see the *interaction* between parameters. This is a *dynamic* representation, meaning that while the central parameter score remains the same throughout the process, one can add secondary

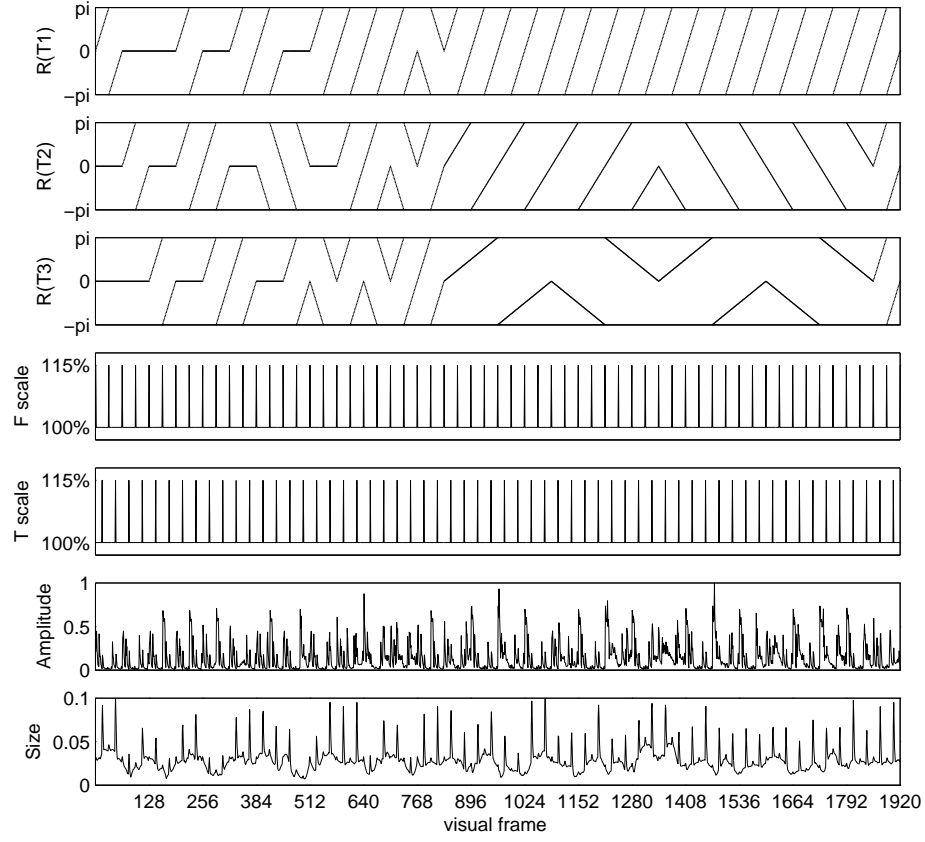


Fig. 6. Score of parameters. $R(T_n)$ is the rotation factor for each transformation. F scale is the scale factor for the entire figure. T scale is the scale factor for the transformations. Amplitude is the power level in each frame of the generated sound file, an extracted parameter. Size is the number of non-black pixels in each frame of the generated animation, an extracted parameter.

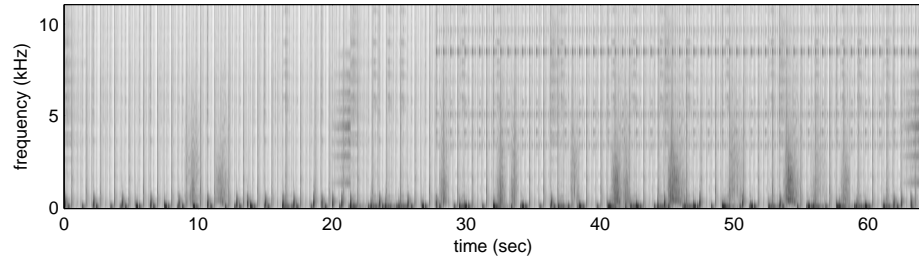


Fig. 7. Fourier spectrum of the sound track generated using the parameter mappings.

parameters, such as the extracted parameters discussed above, and investigate and understand the interaction between the central parameters and these new extracted parameters. It is not necessary to show or describe the mapping that generates these secondary parameters, since the original central parameters and the new extracted parameters are both present in the representation. The parameter set mapping is inherent in the representation.

The resonance inherent in the central parameter variation has been emphasized in both the music and animation, with the strong beats in the music and the scaling applied to the animation. The beats present in the central parameters also show up as beats in the secondary parameters.

5 Conclusions

The motivation for this work was the development of a process by which a single, central, parameter variation could be used to develop intimately related music and animation. Our initial experience with this approach has been very encouraging. Long-term success with this approach is only attainable with a highly developed computer-support structure, similar to the one presented here, which allows the musician and animator to exercise their judgment while at the same time sparing them from the details of the parameters being explored. In future, we will look at ways to provide musicians and animators with more ready access to the parameterizations and schemas put forward by various artists, Schillinger [?] and Whitney [?] included. Conceptually, Whitney's ideas have meshed well with the approach we have presented here.

The use of parameter spaces by artists is directly related to the control they have over the parameters. The systematic exploration of parameter spaces as described here allows the artist to pass on some of the "grunt work" of navigating the space of alternatives for composition to the computer. The system does not limit the artist's creativity in any way, but it also does not require the artist to examine every permutation and possibility that is contained within the parameter space. This entails a great deal of balance while removing tedium without increasing restrictions. An exhaustive search is impossible, but a fully constrained search is uninteresting, and risks missing desired areas of the parameter space. The artist using the system can therefore discover global areas of interest and iteratively constrain the search space to extract more and more detail, with the consistent option of re-tracing and examining other areas. The goal is to enhance the composition process by enabling exploration based on the artist's conceptualization of the problem, which is translated into an appropriate parameterization of the problem space.

The discovery of appropriate parameter mapping paradigms is another issue which depends on the situation. In some mediums, there exist well known and well used parameterizations, and these should be utilized whenever possible. This allows the artist to continue to use paradigms with which he or she is familiar, and perhaps examine them in a new light, but again, they should not be used to artificially constrain the exploration.

Further work will include the investigation of which parameters artists will use within the context of this type of system, and how fully they are able to make use of the extended abilities such a system could give them. To this end, we will prepare and execute perceptual experiments to ascertain the interaction modes that are most helpful to the collaborative composition process, since part of the goal of this work is to make composition more accessible in general. Interfaces will be developed and examined to free composition from the domain of experienced composers. It should be noted that this does not introduce a limit on virtuosity, and experienced and insightful artists should in no way be threatened by the ability of amateur artists to explore and compose.

6 Acknowledgments

Matthew McKague and Tim Maciag have helped to develop the system to its current state. This work was supported by the University of Regina and the Natural Sciences and Engineering Research Council of Canada.

References

1. Apple Computer. *iTunes*. Online: [<http://www.apple.com/itunes/>], Retrieved January 29, 2004.
2. Apple Computer. *Soundtrack*. Online: [<http://www.apple.com/soundtrack/>], Retrieved January 29, 2004.
3. R. M. Baecker, J. Grudin, W. A. S. Buxton, and S. Greenberg. *Readings in Human-Computer Interaction: Toward the Year 2000*. Morgan Kaufmann, 1995.
4. J. Bertin. *Graphics and Graphic Information-Processing*. de Gruyter, 1981. Translated by W. J. Berg and P. Scott.
5. M. A. Boden. Creativity. In M. A. Boden, editor, *Artificial Intelligence: Handbook of Perception and Cognition*, chapter 9, pages 267–291. Academic Press, 2nd edition, 1996.
6. B. Eno and P. Schmidt. *Oblique Strategies: One Hundred Worthwhile Dilemmas*. Online: [<http://www.rtqe.net/ObliqueStrategies/>], Retrieved January 29, 2004.
7. J. Foley and W. Ribarsky. Next-generation data visualization tools. In L. Rosenblum et al., editor, *Scientific Visualization*. Academic Press, 1994.
8. Michael Good. MusicXML for notation and analysis. In W.B. Hewlett and E. Selfridge-Field, editors, *The Virtual Score*, pages 113–124. MIT Press, Cambridge, 2001.
9. J. Hahn, J. Geigel, J. W. Lee, L. Gritz, T. Takala, and S. Mishra. An integrated approach to motion and sound. *Journal of visualization and computer animation*, 6(2):109–123, 1995.
10. D. H. Hepting. Interactive evolution for systematic exploration of a parameter space. In C. H. Dagli et al., editor, *Intelligent Engineering Systems through Artificial Neural Networks*, volume 13, pages 125–131. ASME Press, 2003.
11. D. H. Hepting, F. D. Fracchia, J. C. Dill, and R. D. Russell. Cogito: a system for computer-aided visualization. Technical Report CMPT TR 96-02, Simon Fraser University, 1996. Unpublished technical sketch presented at SIGGRAPH 96.

12. D. H. Hepting, P. Prusinkiewicz, and D. Saupe. Rendering methods for iterated function systems. In H.-O. Peitgen, J. M. Henriques, and L. F. Penedo, editors, *Fractals in the Fundamental and Applied Sciences*, pages 183–224, New York, 1991. North-Holland. Reprinted in *Fractals: from Folk Art to Hyperreality, SIGGRAPH 92 Course #12 Notes.*, Edited by P. Prusinkiewicz. ACM SIGGRAPH, New York, 1992, pp. 3-1 to 3-41.
13. Y. E. Ioannidis, M. Livny, S. Gupta, and N. Ponnekanti. Zoo: A desktop experiment management environment. In *Proceedings of the 22nd International VLDB Conference*, pages 274–285, 1996.
14. Joseph Wayand John G. Neuhoff, Gregory Kramer. Sonification and the interaction of perceptual dimensions. In *ICAD*, 2000.
15. S. Kochhar, J. Marks, and M. Friedell. Interaction paradigms for human-computer cooperation in graphical-object modelling. In S. MacKay and E. M. Kidd, editors, *Proceedings of Graphics Interface '91*, pages 180–189, 1991.
16. W. Lytle. Animusic: a computer animation video album (DVD), 2001.
17. Wayne Lytle. *Computer Animated Music*. Online: [<http://www.animusic.com/software.html>], Retrieved January 29, 2004.
18. B. B. Mandelbrot. *The Fractal Geometry of Nature*. W. H. Freeman, New York, 1982.
19. J. Marks et al. Design Galleries: A general approach to setting parameters for computer graphics and animation. In *SIGGRAPH '97 Conference Proceedings*, pages 389–400, 1997.
20. W. Moritz. Mary ellen bute: Seeing sound. *Animation World Magazine*, May 1996.
21. D. A. Norman. *The Psychology of Everyday Things*. Basic Books, New York, 1988.
22. Nullsoft. *Winamp*. Online: [<http://www.winamp.com/plugins/>], Retrieved January 29, 2004.
23. D. N. Perkins. Insight in minds and genes. In R. J. Sternberg and J. E. Davidson, editors, *The Nature of Insight*, pages 495–533. MIT Press, Cambridge, MA, 1995.
24. J. Schillinger. *The Schillinger System of Musical Composition*. Carl Fischer, 1946.
25. H. Simon. *Models of Discovery*. Reidel, 1977.
26. B. Spinello. Notes on “Soundtrack”. *Source - Music of the Avant Garde*, 1970.
27. K. Stockhausen. *Towards a Cosmic Music*. Element Books, 1989.
28. Russell Storms. Auditory-visual cross-modal perception. In *ICAD*, 2000.
29. T. Takala and J. Hahn. Sound rendering. In *Computer Graphics: SIGGRAPH '92 Conference Proceedings*, pages 211–220, 1992.
30. J. Whitney. *Digital Harmony: On the Complementarity of Music and Visual Art*. Magraw-Hill, 1980.