

A SYNAESTHETIC APPROACH FOR A SYNTHESIZER INTERFACE BASED ON GENETIC ALGORITHMS AND FUZZY SETS

Günther Schatter, *schatter@medien.uni-weimar.de*
Emanuel Züger, *zueger@medien.uni-weimar.de*
Christian Nitschke, *nitschke@medien.uni-weimar.de*
Bauhaus University Weimar, Faculty of Media, Germany

ABSTRACT

This paper presents the concept and implementation of a graphical user interface for the generation of electronic sounds with a synthesizer. We have successfully implemented a complex system of 3D-computer graphical interfaces controlled by several MIDI-devices for a synaesthetic architecture. The processing of the sound defining parameters using fuzzy sets and genetic algorithms in a learning environment made this approach possible. We show how the electronic music system lowers the difficulties to generate target sounds in a purposeful manner.

1. INTRODUCTION

The range of electronic principles and tools for music production is an exciting field and has become very large. Much efforts were made developing new principles of sound synthesis and creating new devices as well as controllers translating motions into MIDI signals that control sound generating devices. This development has exposed the problems of a generally poor usability of electronic musical instruments, in particular of synthesizers [1]-[5]. However, for the control of modern synthesizers several dozens of switches, knobs and faders are available. But similar keys and buttons will not respond with the same result and vice versa. Hence, providing novices and children with modern sound systems, enabling them to compose or play music in an intuitive manner arises as a challenging task.

Our approach is inspired by synaesthetic concepts: A certain perceptual mode is linguistically related with terms belonging to a different perceptual mode by association or connotation. Such well-known metaphors are *warm sounds* or *cold colors*; most common are *colored sounds*. Properties of sound should be described by visual models or metaphors enabling a *user-individual* mapping, memorization and reproduction of sounds. Significant acoustic attributes are multisensually mapped on corresponding terms of visual perception. In practice a 3D-model which describes an electronically generated sound can be manipulated to obtain a certain acoustic event. This mapping neither is causal nor free of contradiction: it is highly subjective. So this approach needs personalization. Since each human has an individual perception, there

are provided tools for free personalization of the graphical 3D-user-interface.

2. DESCRIPTION OF THE SYSTEM

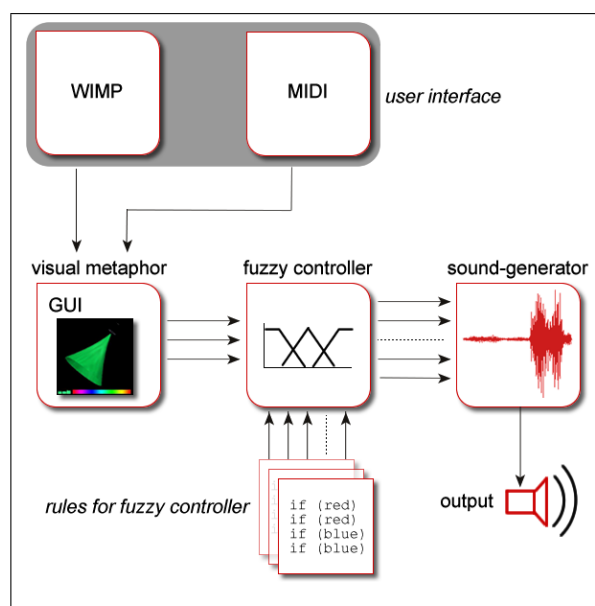


Figure 1. System overview of the *Synthesizer*.

The sound-shaping 3D-metaphor is controlled by the use of a WIMP-interface (window, icon, mouse, pointer) or MIDI-capable external interfaces. Therefore unconventional input devices allow interesting and new opportunities for live-performances. The control of pitch, rhythm, melody, level and localization is preferably done via MIDI-interfaces (Figure 1). The whole system consists of two software components: a sound-generator for practical use as a music instrument (*Synthesizer*) and a system for personalization of the *Synthesizer* (*Assistant*).

3. IMPLEMENTATION

3.1. Synthesizer

The *Synthesizer* is implemented as a VST-Plugin and consists of three parts: Sound generation by subtractive synthesis, 3D-visualization as graphical user-interface and a

set of fuzzy-controllers for the mapping between user-interface and sound-generator. The subtractive sound-generator consists of basic algorithms and provides two oscillators capable of generating sine-, sawtooth- and pulsewaves. Furthermore there is implemented a random gaussian noise, two ADSR-envelope-generators, one filter (low-, high- or bandpass, 4-pole with resonance) and one amplifier. This results in the sound-generator having 23 independent parameters.

The visualization and therefore the user-interface consists of three abstract visual input-parameters: *Form*, *Material* and *Color*. Visualization of the 3D-model is done using a slightly modified OpenGL-API in order to work within the VST-framework. The modifications were derived from the MIVI-project [10].

The mapping of the parameters of the sound-generator onto the very abstract visual metaphor was the essential challenge: We had to figure out how the 23 parameters of the sound-generator can be reduced simultaneously and comprehensibly to five parameters of the visual metaphor. We had to keep in mind, that this reduction underlies the subjective perception of each user. Therefore we use knowledge-based mapping with fuzzy-controllers which are adaptable to each user. Due to this we take advantage of the Free Fuzzy Logic Library [11], where plain ASCII-files store fuzzy rules and invoke them at runtime.

Material and *Color* represent the static aspects of a sound. *Material* is visualized as texture and represents the very basic aspects of a sound: soft, neutral or rough. In order to deal with *Color*, we brought up the principle of a *water-tap*: It ranges from blue (cold sound) to red (warm sound). The most flexible input-modality is the *Form* of the 3D-model which influences the envelope-generators and thus the behavior of the sound over time, the most complex of all sensations. The *Form* is derived from the envelope of the sound-output and provides three independent parameters: *Height*, *Width* and *Bulb* which are controlled by two visual tools (Figure 2). The mapping between *Form* and envelope-generator is done either linear or following rules of fuzzy logic.

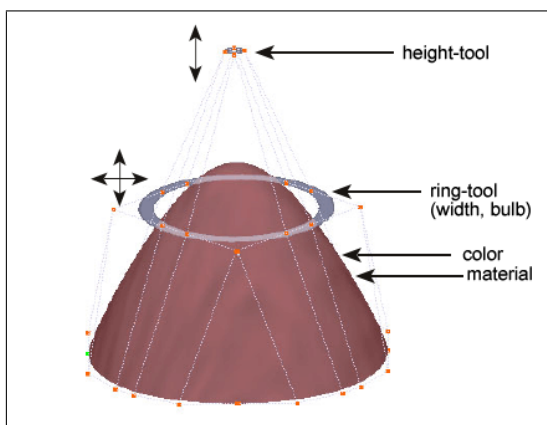


Figure 2. 3D-model, providing five independent parameters.

3.2. Personalization of the 3D user interface

The personalization of the user-interface turned out to be a critical part. Hence the *Assistant* was developed as an additional software. It aids the user in assigning best matching generated sounds to some preselected metaphors. These assignments represent the mapping between the 3D-model and the sound-generator, being the knowledge-base for computing fuzzy controllers accordingly.

For the personalization we employ two distinct strategies: On the one hand in a *manual* operation mode a technical experienced user should be able to establish the assignments without reducing the diversity of parameters (Figure 3). On the other hand in an *automatic* operation mode a less experienced user should be faced with less complexity. Therefore we employ a genetic algorithm as an iterative optimization technique to find the best approximations of the *result sounds* to the *target sounds* given by the user.

As metaphors to be processed we chose the ones that reside in the corners of the two-dimensional space which is spanned by the parameters color and material as well as minimum and maximum of the parameters height, width and bulb. So, for not overstraining the user a compromise was found between the number of metaphors and the resulting computation time needed for the whole personalization process.

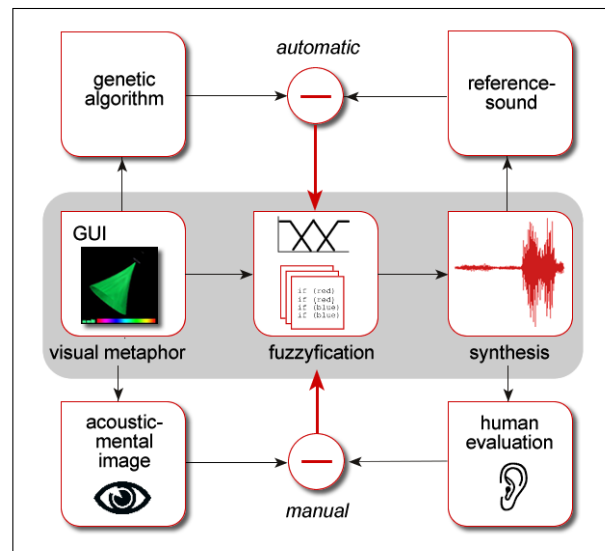


Figure 3. Operation modes of the *Assistant*.

The personalization of the *Synthesizer* consists of these steps:

1. The 12 visual metaphors are presented to the user (Figure 4). For each one an approximating result sound has to be determined. The task is accomplished either manually or automatically, depending on the musical/technical background of the user.
2. In this process the whole parameter-space is exploited to create suitable fuzzy-controllers in the

form of FCL files which are further used by the *Synthesizer*.

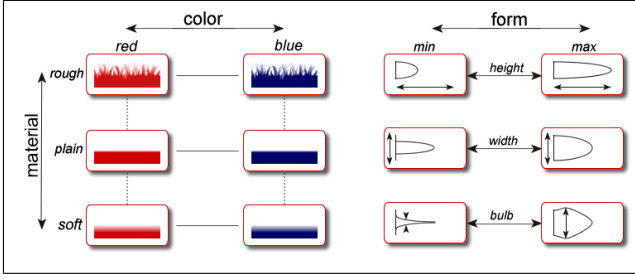


Figure 4. The 12 metaphors used by the *Assistant* for the personalization of the *Synthesizer*.

In manual operation mode the *Assistant* is used just to record the parameter-input of the user, followed by the generation of the fuzzy controllers. In automatic mode the *Assistant* has to find a parameter combination and thus the output of the sound-generator, which approximates the target-sound as much as possible.

The set of possible result sounds is given by the set of parameter combinations $C = P_1 \times \dots \times P_{23}$ which is defined as the cross product of the sets of the parameter values P_i ($i = 1 \dots 23$). This set represents the whole parameter-space. The quality of any combination c denotes its distance to the target sound. This is determined through the evaluation function $f : C \rightarrow \mathbb{R}_+$. While the aim is to select an optimal combination $c^* \in C$ as the result sound so that $f(c^*) \leq f(c) \forall c \in C$ holds, we result in a combinatorial optimization problem $P = (C, f)$.

In the case of the employed algorithm for subtractive sound synthesis a solution space of $C = 5.4 \cdot 10^{44}$ sounds arises. An application of brute-force techniques, performing an evaluation of all possible solutions, would result in an unacceptable increase of computation time. So we decided to use a genetic algorithm [12]. The task can be abstracted in terms of the genetic algorithm. A result sound (individual) is represented by its unique parameter combination c (genome). The optimization (evolution) is realized employing operators that are adapted from the natural principles of *selection*, *crossover* and *mutation*. The probability of selecting an individual is determined by its fitness value which is returned from the evaluation function $f(c)$.

Beside the number of passed generations we also employ a convergence test as termination criterion. It checks up whether in a certain period of time a given improvement in fitness could be achieved. If not, convergence to an optimal fitness value is assumed and the algorithm terminates. This is necessary since the achievable fitness value depends on the ability of the sound-generator to approximate the particular target sound. The fitness function $f(c)$ of any suggested result sound is calculated using a distance measure to the target sound (Figure 5). First of all a difference spectrum is computed from the spectra of both sounds. According to principles of psychoacoustics, a weighting and irrelevance reduction is done

on both spectra. The lower the difference spectrum, the higher the fitness value of the respective result sound will be. The function is defined as:

$$f(c) = \frac{\sum_{i=1}^{FFT_windows} \sum_{j=1}^{FFT_bands} sf |v(t)_{ij} - v(c)_{ij}|}{\max[num_samples(t), num_samples(c)]}$$

where t denotes the target sound and $sf \in [0, 1]$ a psychoacoustic weighting factor based on the human audible curve. $v(t)$ and $v(c)$ indicate the FFT spectra of target and result sound.

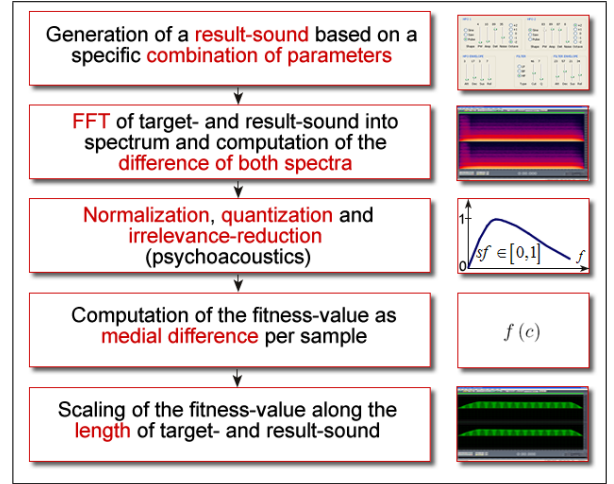


Figure 5. Determination of the fitness value of a result sound in automatic operation mode.

For the evaluation of the automatic operation mode we employed an experimentally derived configuration for the genetic algorithm: 10 populations with 70 individuals each were running at the same time. 2-point-crossover and flip-mutation with respective probabilities $p_c = 0.9$ and $p_m = 0.03$ were used. The maximum number of 500 generations was defined, whereas after every 50 generations a convergence test was made.

4. RESULTS

The first challenges for providing a new and exciting user-interface for sound-synthesis were accomplished with the integration of 3D-visualization within the VST-framework as well as the use of non-linear mapping from few to many parameters using fuzzy-logic. With the proposed techniques we solved this task while achieving the conditions of typical music production environments: For example, on a machine with Pentium 4 3GHz-CPU, 512MB DDR-RAM and OpenGL-accelerated graphics-card we achieve up to 16 *Synthesizers* running at the same time without loss of synchronisation and sound-quality using Steinberg Cubase SX 2.0.

The second challenge was to evaluate and optimize the 3D-visualization: This led to important questions:

1. Is it possible to distinguish sounds significantly using the metaphors *Form*, *Material*, and *Color*?

2. How much are these metaphors influenced by the user's individual perception?

Evaluations were made which turned out that the visual metaphors could not be employed as an exclusive controlling strategy for the *Synthesizer*. There is the need for an individual personalization of the user-interface for each user. This particular problem led us to the development of the *Assistant*.

Experienced users are able to use the *Assistant* in manual operation mode. The automatic operation mode turns out to be useful if the sound-generator is able to approximate a target sound quite well. At the end the quality of the result sounds depends on the sound generation technique as well as the kind of target sound. Complex, in particular impulsive, target sounds do not yield satisfying results.

User evaluation was done in a two-stage set of tests which consists of generation and verification. In the first step expert users had to use the *Assistant* in manual operation mode. In the second step the previously derived metaphor sound assignments should be verified by the participants of the test. The result sounds acquired during the configuration by the *Assistant* with expert users were presented several times in mixed order to other participants, whereby they had to classify them into projections of the metaphor space.

The hypothesis of an overall multisensual conjunction between *Color*, *Material* and sound can only be confirmed partially. The *Form*-parameters controlling the temporal invariant part of the sound turned out to be a problem. The thesis which states a perceptual conjunction between shape and sound does only barely hold. It turned out in particular that the effects of adjusting the metaphors *Width* and *Bulb* are not intuitively understood by the users.

The complexity of the presented questions prevents us from giving a general conclusion for the interaction of visual and acoustic parameters.

5. CONCLUSIONS

We developed a modular software which combines virtual analog sound-synthesis, 3D-enabled user-interface and machine-learning for optimization and personalization. This learning-system employs genetic algorithms, digital signal processing, fuzzy logic and several techniques of sound-generation. We think that the personalization of electronic musical instruments is a new and interesting approach for the creation of intuitive and responsive sound toys and music devices [13]. Actual computing-technologies offer a rich potential to fulfill these ideas based on artificial intelligence and smart devices.

6. REFERENCES

- [1] Jordà, S. *FMOL: Toward User-Friendly, Sophisticated New Musical Instruments*. Computer Music Journal, 26(2002)3, pp. 23-39.
- [2] Cook, P. *Principles for Designing Computer Music Controllers*. <http://www.cs.princeton.edu/~prc/CHI01Web/prcchi0s1.pdf> (acc. 23.02.2005).
- [3] Farbood, M.; Pasztor, E.; Jennings, K. *Hyperscore: A Graphical Sketchpad for Novice Composers*. IEEE Computer Graphics and Applications, 24(2004) Jan./Feb., pp. 50-54.
- [4] Wanderley, M.; Orio, N. *Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI*. Computer Music Journal, 26(2002)3, pp. 62-76.
- [5] Mulder, A. G. *Design of Virtual Three-dimensional Instruments for Sound Control*. Doctoral thesis Groningen, 1989.
- [6] Seago, A.; Holland, S.; Mulholland, P. *A Critical Analysis of Synthesizer User Interfaces for Timbre*. Technical Report 2004/21, The Open University Milton Keynes.
- [7] Rolland, P.-Y.; Pachet, F. *A Framework for Representing Knowledge about Synthesizer Programming*. Computer Music Journal 20(1996)3, pp.47-58.
- [8] Miranda, E. R. *At the Crossroads of Evolutionary Computation and Music: Self-Programming Synthesizers, Swarm Orchestras and the Origins of Melody*. Evolutionary Computation 12(2004)2, pp. 137-158.
- [9] Leman, M. *Visualization and Calculation of the Roughness of acoustical Musical Signals using the Synchronization Index Model*. Proceedings DAFX-00, Verona, 2000.
- [10] University of York. *MIVI - a musical instrument visual interface*. <http://www.nashnet.co.uk/mivi/index.htm> (acc. 23.02.2005).
- [11] *Free Fuzzy Logic Library*. <http://ffll.sourceforge.net> (acc. 23.02.2005).
- [12] Mitchell, M. *An Introduction to Genetic Algorithms*. Bradford Books, MIT Press, 1999.
- [13] Schatter, G.; Züger, E.; Nitschke, C.; Pradella, M.; Linke, D. *Intuitive Graphical User Interfaces for the Electronic Sound Generation (in German)*. Proceedings Tonmeistertagung 2004, Leipzig.