Usual sentence derivation from a grammar will automatically generate the desired structure for the generated music. The user may impose restrictions, e.g. on the maximum number of fragments, of fragments of a given type, of repetitions of certain sequences of types. The expected result is the automatic generation of music with sound similar in quality to the previous ones, and showing the structure created by the system under user's specification.

6.4-Phase 3 - to be started in short term - the next stage refers to obtaining structuring rules automatically. The underlying grammar of the structure imposed to automatically generated compositions will be itself deduced by the system. A musician must help in this activity by supplying structural analyses for a significant set of works to be used as models, identifying and classifying the fragments in each work. Then, an algorithm is executed to make a grammatical inference that generalize the information extracted from the given samples, in order to infer a formation rule to be used as a grammar describing the desired structure. This phase will result in a complete synthesis program, for music will then be obtained automatically both in contents and in formal structure. User's participation will be restricted to inserting information on the structure of the samples used as models.

6.5-Phase 4 - this phase is foreseen to begin in some years. It is much more complex, because it should include heavier artificial intelligence resources; in order to operate with no human-supplied information concerning the structure of the pieces used as models. Structural analysis should be made by programs with enough built-in knowledge on musical analysis to allow automatic identification of the musical elements present in the score. This phase will produce software that will be able to perform musical analysis and synthesis, without direct participation of the user.

7-References

COPE, DAVID. **Experiments Musical in Intelligence (EMI): Non-linear Linguistic-Based Composition** Interface, vol, 18, p. 117-139, 1989.

COPE, DAVID. **Computer Modeling of Musical Intelligence in EMI**. Computer Music Journal, vol 16, no. 2, Summer, p. 69-83, 1992.

COPE, DAVID. **Music & Lisp**. OH Expert, March, 1988, p.26-33.

DODGE, CHARLES; JESSE, THOMAS A.. **Composition With Computers - Computer Music - Synthesis, Compositions and Performance.** NY, Schirmer, London, Coller MacMilan, 381 pg., p.265-321, 1985.

GOGINS, MICHAEL. Interated Functions Systems Music. Computer Music Journal, vol. 15, no. 1, Spring, 1991, p.40-48

HUDAK, PAUL; MAKUCEVICH, TONE; DDE, SYANGA; WONG,BO.**Haskore Music Notation–An Algebra of Music** -. Journal of Functional Language, vol 6, May, 1996 p.465-483

LIMA, LUCIANO VIEIRA. **Um Sistema de Composição Musical Dirigido por Estilo.** Tese de Doutorado, Universidade de São Paulo, São Paulo - Brasil, 1998.

SMOLIAR, STEPHEN W. Book Review and Response - **Models of Musical Communication and Cognition**. Interface, vol. 18, 1990, p. 361-371

TODD, PETERSOM. **Connectionist Approach to Algorithmic Composition**. Computer Music Journal, vol. 13, no.4, Winter,1989, p.27-43.

SOUND FUNCTORS APPLICATIONS

JÔNATAS MANZOLLI[2], ADOLFO MAIA Jr.[1,2]

Mathematics Department[1]

Interdisciplinary Nucleus for Sound Communications (NICS)[2]

University of Campinas (UNICAMP) - Brazil

Jonatas@nics.unicam.br, adolfo@nics.unicamp.br

ABSTRACT

The concept of mathematical structures called Functors can be useful to develop a large number of compositional algorithms. We introduce here concepts such as Categories and Functors as generalised sound construction tools. We define two applications: a functor between plane curves and sound events based on the MIDI protocol and a functor between the class of continuous curves $C(x)$ *defined in a finite interval* $I \subset \Re$ *and the class* Ψ *of Fourier Spectra* $A(\nu)$.

INTRODUCTION

Music composition is plenty of procedures that can be related to mathematical symmetries, it is presented in (Hofstadter 1989), a study relating music, design and mathematical structures. On the other hand, music symmetries can be better understood and handled if we search for underlined or hidden mathematical structure that generates them. In this work we show that the intuitive use of symmetry and associations in music can be studied and explored through the mathematical concepts of Category and Functor.

This approach is interesting, not only because it can be used to classify sound material, but mainly for it furnishes new relations pointing out to an enormous variety of sound generation methods. In short, we claim that Functors, in the same way it was firstly devised in mathematics (Maclane & Birkohoff 1953, 1979; MacLane 1971), can be generalised as universal tools to construct mathematical models for music composition and sound synthesis.

There has been a series of approaches applying mathematics to build sound structures such the use of 1/f noise fractal distribution (Voss & Clark 1978; Bolognesi 1983), non-linear dynamical systems and iterated function systems (Pressing 1988; Scipio 1990; Gogins 1991). There is a study about these systems in (Manzolli, 1993a).

Our group has been worked with applied mathematics to produce sound design machines focusing new methods for sound synthesis using non-linear dynamics (Manzolli 1993b), mathematical models for algorithm composition such as Markov Chains and

Boundary Functions (Manzolli & Maia 1995) and development of interactive desktop and gesture interface for composition in real time (Manzolli & Ohtsuki 1996).

Below, we discuss Functors as generalised Algorithmic Compositional tools for sound construction defining Class, Morphism, Category and Functor. It follows applications in the sound domain.

CATEGORIES AND FUNCTORS

Functor is a kind of function, as suggested by its own name, but it is not an ordinary one because it carries the underlined structures of the sets in which it is applied. These sets are called Categories. More precisely, a **Category** Φ is defined by three proprieties:

a) A *class of objects* **A, B, C...**

b) For each pair of objects $\mathbf{A, B} \in \Phi$ we have a set of applications (morphisms) **M(A,B)** from **A** to **B**.

c) For each triple of objects $\mathbf{A, B, C} \in \Phi$ we have a composition law for the morphisms

$$\mathbf{M(A,B) \times M(B,C) \rightarrow M(A,C)}$$
$$(f,g) \longrightarrow g \circ f$$

Now, for the proprieties above it follows three axioms, which are primary properties of these categories:

A_1). The sets of morphisms **M(A,B)** and **M(C,D)** are mutually disjoints unless $\mathbf{A = C}$ and $\mathbf{B = D}$.

A_2) Associative Law: $\mathbf{h(gf) = (hg)f}$.

A_3) Existence of Identity: for each object **A** there exists a morphism identity $\mathbf{1_A}$: $\mathbf{A \longrightarrow A}$ such that for any $\mathbf{f: A \longrightarrow B}$ and $\mathbf{g: C \longrightarrow A}$ we have $\mathbf{f \circ 1_A = f}$ and $\mathbf{1_A \circ g = g}$.

Usually, the theory of categories and functors is mathematically involved. Here we use only basic properties of the definition above to show that underlined structures can be mapped from categories of mathematical objects to categories of sound objects and the last undertake formal properties of their mathematical counterparts via a functor.

Given two categories Φ and Ψ, a functor **F** between Φ and Ψ is a map which associates each object $\mathbf{A} \in \Phi$ to an object $\mathbf{F(A)} \in \Psi$

$$\mathbf{F}: \Phi \longrightarrow \Psi$$
$$\mathbf{A \longrightarrow F(A)}$$

and for each morphism $\mathbf{f} \in \mathbf{M(A,B)}$ associates a morphism $\mathbf{F(f)} \in \mathbf{M(F(A), F(B))}$ with the properties

$$\mathbf{F(gf) = F(g)\,F(f)}$$
$$\mathbf{F(1_A) = 1_{F(A)}}$$

Functor **F** operates on morphisms as well on elements of the category Ψ. In this way, the structure of a product (or composition) between two morphisms in the category Φ is transported to morphisms in the category Ψ via functor **F**.

There are too many examples of mathematical functors. Since we are focusing music application only, two simple examples of functors applied to sound categories are presented in the next section.

APPLICATIONS

EXAMPLE 1 - **Sequence of MIDI Events**

We begin with the mathematical category Φ = {***continuous finite curves in a bounded region*** $\mathbf{U} \subset \Re^2$}. Given two curves $\mathbf{C_1, C_2} \in \Phi$, a morphism in Φ as an application

$$\mathbf{f: C_1 \longrightarrow C_2}$$
$$\mathbf{x \longrightarrow f(x)}$$

which means to deform $\mathbf{C_1}$ into $\mathbf{C_2}$, i.e. $\mathbf{f} \in \mathbf{M(C_1, C_2)}$. The product **gf** is defined as the composition **gf = gof**

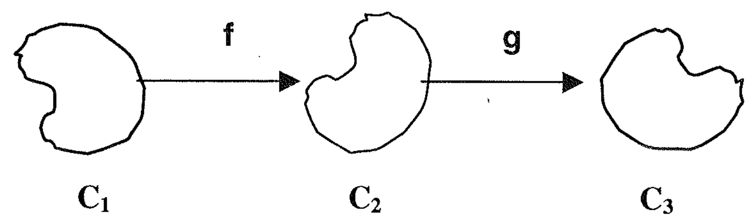

Figure 1. Composition of Plane Curves

We define Ψ as a category of functions to control sound parameters described by the MIDI (Music Instrument Digital Interface) protocol. Thus, $\mathbf{F(f)} \in \mathbf{M(F(C_1), F(C_2))}$ and **F(f)** deforms $\mathbf{F(C_1)}$ to $\mathbf{F(C_2)}$. Given a curve **C** in Ψ, the function $\mathbf{F(C)} \in \Psi$ can be defined in several different manners. Particularly, **F(C)** can be expressed by the Distance Function between a fixed point in the plane to the points of the curve **C**. It is easy to see that the functor property is satisfied, namely

$$\mathbf{F(g(f(C))) = F((gf)(C)) = F(g)F(f)(F(C))}$$

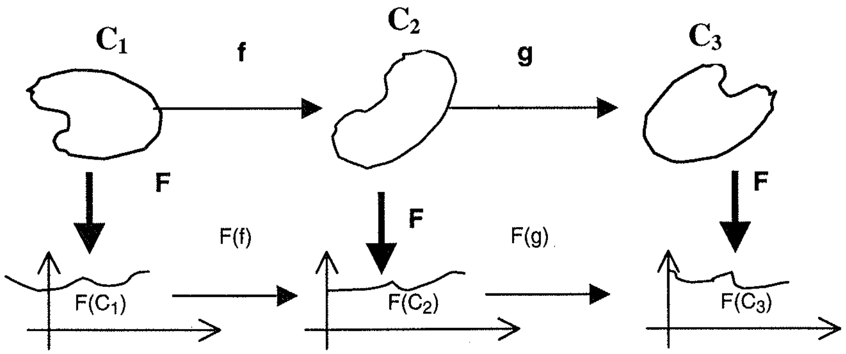

Figure 2. Diagram of the Functor operation

We have implemented this functor of plane curves to MIDI parameters in a computer program called CurvaSom. This software uses a graphic interface running in a MS Windows NT environment. Here we are just introducing main theoretical aspects of the Sound Functor approach.

EXAMPLE 2 - **Sound Synthesis**

In this example, an application of a functor is used to build Spectral Envelops. We take as mathematical category, a class Φ of smooth functions $\mathbf{C(x)}$ defined in a finite interval $\mathbf{I} \subset \Re$ and as sound category, a class Ψ of Fourier Spectra $\mathbf{A(\nu)}$ of a finite set of sounds. Here the function $\mathbf{C(x)}$ acts as a shaper of a spectral set. Starting from a fixed spectrum, which we call Input Spectrum, we use deformation of curves as morphisms in both classes of functions. We construct the following Functor:

$$\mathbf{F}: \Phi \longrightarrow \Psi$$
$$\mathbf{C} \longrightarrow \mathbf{F(C) = C(A(\nu))}$$

Where the parameter $x = A(\nu)$*, with* $0 \leq x \leq A_{max}$ *and* A_{max} *is the maximal amplitude considered.*

The morphisms in Φ are deformations of smooth functions and the morphism in Ψ can be chosen as $\mathbf{F(f)=f}$. This means that the morphisms in Φ are used as spectral deformations. Using the above definition it is easy to see the property $\mathbf{F(g)\ F(f) = gf = F(gf)}$ is satisfied.

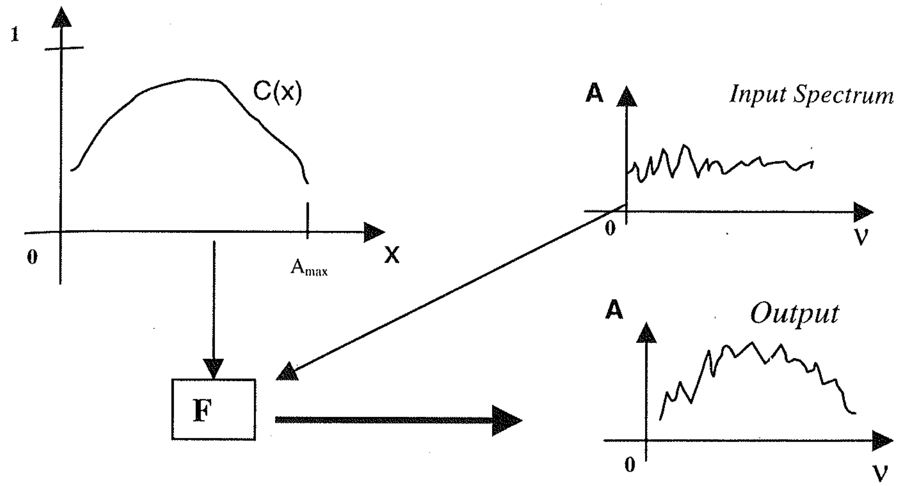

Figure 3. Spectrum Functor Diagram

CONCLUSION

We can construct several examples similar to the above ones. Another example using a different mathematical construction is presented in (Maia, Valle & Manzolli 1998). Other mathematical structures like groups, lattices, algebras, several different geometrical and topological structures can be used as mathematical categories (see MacLane 1971) in order to get sound outputs through a suitable choice of functors. Functors allow us to construct a huge quantity of sound outputs reflecting the structure (or symmetry in several cases) of external mathematical categories used to generate them. Musicians who work on Computer Music, aware of the properties of functors, can expand their sound tools in a way, certainly, not yet explored by the authors of this paper.

Mathematical tools in composition brings new possibilities for composers envision development of Compositional Systems. Research in Computer Music provides powered tools for constructing these systems. This union of artistic and mathematical knowledge creates a framework for investigation and music production, an environment for applying mathematics in order to manipulate sound structures. Mathematical models presented here could be expanded using graphic interfaces to create new musical performance situations, making mathematical design to produce computer music instruments.

REFERENCES

Bolognesi, T. (1983). Automatic Composition: Experiments with Self-Similar Music. *Computer Music Journal* 7(1):25-36.

Gogins, M. (1991). Iterated Functions Systems Music. *Computer Music Journal* 15(1):40-48.

Hofstadter, D. R. (1989). *Gödel, Escher, Bach: an eternal golden braid.* Vintage Books, New York, ISBN 0-394-75682-7.

MacLane, S. (1971). *Categorias for the working mathematician.* New York, Springer-Verlag.

MacLane, S. & G. Birkhoff (1953). *A Brief Survey of Modern Algebra.* New York, the MacMillan Company.

MacLane, S. & G. Birkhoff (1979). *Algebra.* New Yok, Macmillan Company.

Manzolli, J. (1993a). Non-linear Dynamics and Fractals as a Model for Sound Synthesis and Real Time Composition. PhD Thesis at the University of Nottingham, England.

Manzolli, J. (1993b). *Musical Applications Derived From FracWave Sound Synthesis Method. Proceedings of the 94th Audio Engineering Society Convention*, Berlin.

Manzolli, J. (1994). FracWave: Non-linear Dynamics as Timbral Constructs. Procedings of the *of the XIV Congress of the Computer Brazilian Society, I Symposium on Computer Music,* Caxambú - Brasil.

Manzolli, J. & A. Maia Jr (1995). Interactive Composition Using Markov Chain and Boundary Functions. *Procedings of the XV Congress of the Computer Brazilian Society, II Symposium on Computer Music*, Canela, Brazil.

Manzolli, J. & W. Ohtsuki (1996). INTERASOM: a desktop for Algorithmic Composition. *Procedings of the XVI Congress of the Computer Brazilian Society, III Symposium on Computer Music,* Recife, Brazil.

Maia Jr., A., R. do Valle & J. Manzolli (1998). Estruturas Matemáticas como Ferramenta Algorítmica para Composição. Submitted to the XI Congress of the Brazilian Association for Research in Music, ANPPOM, University of Campinas.

Pressing, J. (1988). Non-linear Maps as Generators of Musical Design. *Computer Music Journal* 12(2):35-45.

Scipio, A. (1990). Composition by Exploring of Non-linear Dynamic System. *Proceedings of the 1991 International Computer Music Conference.*

Voss, R. F. & J. Clarke, (1978). 1/f noise music: Music from 1/f noise. *Journal of the Acoustical Society of America* 63(1):258-263.

ACKNOWLEDGMENT

We thank Prof. Raul do Valle for fruitful discussions and the Foundation for Research of State of São Paulo - FAPESP for supporting this work.

Busca e Recuperação de Informação Musical

Ana Miccolis
Universidade Federal do Rio de Janeiro
Coordenação dos Programas de Pós-Graduação em Engenharia - COPPE/UFRJ
Programa de Engenharia de Sistemas e Computação
Caixa Postal 68 511- CEP 21945-270 - Rio de Janeiro - RJ - Brasil
E-mail miccolis@cos.ufrj.br

Abstract

The process of musical composition can be benefit by the use of resources extracted from a database of musical passages. For this purpose, the composer may have access to the sound resources and be able to manipulate them in the compositional process. The traditional musical representation employed at the conventional music compositions allow the composer to retrieve the necessary information, to process it and to restore it in the same representation. The musical data in non conventional composition can't be represented only by the conventional written. In this case, the musician needs an resource that let the composer select music passages based on musical characteristics. One of the goals of information retrieval of musical data is to aid the research in compositional resources, like selection of timbre in a database. The database with this capability might provide the access to all the conventional attributes like alphanumeric data and the musical data. This paper investigates this kind of search and retrieval and which criteria one may consider at the musical information retrieval process.

1 - Introdução

Sistemas de recuperação de informação processam requisições de informações sobre arquivos identificando e recuperando destes certos registros que atendem aos requisitos exigidos. A recuperação de alguns registros em particular depende da similaridade entre estes e as consultas. Esta similaridade pode ser medida através da comparação de valores de certos atributos dos registros com os valores requisitados pela consulta. Em muitos sistemas de gerenciamento de banco de dados os arquivos contêm registros homogêneos, com um conjunto de atributos especificado para caracterizar cada item do arquivo e com valores dos atributos que conseguem descrever univocamente e de forma completa os registros armazenados. Nestas circunstâncias a recuperação de informação depende de uma comparação exata entre os valores do arquivo e os da consulta. O nome de um autor ou intérprete, a data de nascimento, a sua nacionalidade e os títulos das obras musicais são tratados como dados não ambíguos e o processo de recuperação baseado nestes critérios é