MATHEMATICAL STRUCTURE AND SOUND SYNTHESIS

Georg Essl
Deutsche Telekom Laboratories & TU-Berlin
georg.essl@telekom.de

ABSTRACT

How can looking for mathematical structure help us with sound synthesis? This question is at the core of this paper. First we review some past work where attention to various kinds of mathematical structure was employed in the context of physical modeling synthesis. Then we propose a first few steps towards algebraisation of abstract sound synthesis using very basic constructions of abstract algebra and category theory.

1. INTRODUCTION

Mathematics has been used in sound synthesis in many diverse ways for a rather long time. We have used the properties of computations that have auditory consequences in many different way to create and manipulate sounds. At the same time mathematics is also a language that helps us classify a problem with respect to certain properties and hence allows us to understand the problem and reduce it to hopefully more easily manageable pieces.

This paper talk about the interplay of sound synthesis and mathematics with physics as a supporting actor.

First we discuss past and current work where mathematics and sound synthesis overlap — leaning towards the authors own interests and thinking. Then we discuss a first few steps towards algebraisation of sound synthesis and we give some concrete examples of synthesis algorithms that naturally derive from abstract structures.

1.1. Mathematics as a study of structure

The last century has seen a development in mathematics that constitutes a rather significant shift in the outlook and goal of mathematics. This change began already in the 19th century where general properties and their structuring power came to the forefront. Algebraic properties were seen as providing a way of classifying what kinds of computations belong together and explain the properties of certain mathematical scenarios. The concept of a group was crucial in the development of Abel and Galois in the work on the solutions of algebraic equations [1]. Klein's Erlangen Program was really a realization that different types of geometries could as well be described by such algebraic properties of geometric transformations [12]. This was then combined with a drive to axiomatization first by Hilbert and then by a group of French mathematicians working under the pseudonym Nicolas Bourbaki (and also under their own name). At the same time one might argue there has been a certain fork between mathematics as an applicable language for other sciences and pure mathematics. The idea of structure often is more emphasized in the latter. Here we will describe the interplay of mathematical structure and applied problems in sound synthesis to give an indication of its practical usefulness.

2. PHYSICAL MODELING AND MATHEMATICS

Physical modeling developed into a very active field of research with the serendipitous discovery of the Karplus-Strong algorithm for plugged strings and its physical interpretation by Smith [3, 16]. The algorithms are now known as "waveguide synthesis" and have rather stunning properties. They are accurate at sample points and show a computational efficiency which is constant and independent of spatial sampling [16]. Additionally if one models deviations from the ideal case in a lumped fashion, numerical problems are confined and easy to handle.

One may well be amazed at the fact that waveguides are so efficient and well-behaved, especially because finite difference methods also simulate the same situation, but are numerically more sensitive and their efficiency is determined linearly by the spatial sampling.

Essentially waveguides discretize the form of the solution, whereas finite differences discretize the differential operators. Ultimately both integrate the solution over time. The great advantage is that the form of the solution of the ideal string, classically known as the d'Alembert solution has a specific simple form that allows one to avoid computation:

$$y(x,t) = \frac{1}{2}\left(f(x+t) + f(x-t)\right) + \frac{1}{2}\int_{x-t}^{x+t} g(s)\,ds. \tag{1}$$

where $f(\cdot)$ is a function generated by an initial displacement and $g(\cdot)$ is the response to initial velocity. To simplify notation the speed of sound on the string c is rescaled to equal 1. Hence the solution travels left and right, and with boundary conditions we get a wave periodically traveling the length of the string. The simplicity now comes from the fact that keeping track of traveling disturbances that do not chance shape is equivalent to reordering elements and this can be done with a constant number of operations.

Looking at waveguides suggest a certain methodology if one wants to find comparable solution to other situations than are immediately solved by waveguides. The insights and steps are roughly:

- Forms of solutions are not arbitrary.
- Solutions carry certain properties.
- Formulating the solution to highlight structure.
- Separate problem into aspects that are easier to handle.

Next we will look at ways to incorporate these steps in various ways into work on physical models away from the ideal string. The presentation will pick different aspects of the problem and discuss them separately and are hence by this very nature examples of subproblems of a bigger, more complex problem.

2.1. Propagation in Stiff Structures

Waveguides, at least in the idealized case, assume that the speed of propagation of disturbances is constant. Some small perturbations to this are possible and can be modeled by all-pass filters [16]. However a number of sounding structures, like marimba bars, bells or dinner tables, do not have this property. The stiffness of the coupling within the object leads to frequency-dependent propagation of disturbances. So rather than having the speed of sound c be a constant we assume a frequency dependent propagation speed within the media $c(\omega)$.

One approach to this problem is to take this literally and try to implement this locally, meaning that everywhere all frequencies are propagated according to $c(\omega)$. This can be done by replacing unit delays in waveguides with all-pass filters. The problem with this approach is, however, that the desirable properties of waveguides, numerical convenience and computational efficiency are lost. But our guiding philosophy is "preserve desirable structure if possible". Hence we want to preserve the desirable waveguide structure while accommodating the changed behavior. This can be done by discretizing $c(\omega)$ and assuming that within discrete bands propagation is in fact constant. Then one arrives at substructures that have the properties of waveguides. This idea is the essence of banded waveguides [8] and is depicted in Figure 1.

2.2. Abstraction of Structure

Another interesting aspect of considering structure is the possibility of multiple interpretation. One can have one and the same structure but it may be interpreted differently and hence be seen to be in different contexts. An example of this comes about with banded waveguides. Originally banded waveguides were conceived to simulate the physics of structure-borne stiff one-dimensional systems like marimba bars. But once the structure to do the simulation was in place and was set up to preserve the kind of properties we do find interesting, there was the question of the applicability of this structure. More precisely, the basic question arises, can this method be applied outside the original domain of application. Here the part that comes to hold is the realization which part of the final structure is not a strictly necessary feature to retain validity of the simulation. In the case of banded waveguides the original interpretation of the separate waveguide loops is that they essentially occupy the same spatial direction and space. They model behavior that comes about through repeated traversal of one direction in this space. The abstraction of this model is the lifting of this requirement. If we abstract the structure and allow paths to stand for something else than a confined spatial domain we can get to resonant behaviors in more complex spaces. Now closed loops do not stand for confined propagation in physical space but stand for ensemble propagation in more higher-dimensional physical space that repeats itself to form resonances. Given this easing of interpretation it became possible to find physically meaningful interpretations of banded waveguides in two and three dimensions [8, 3].

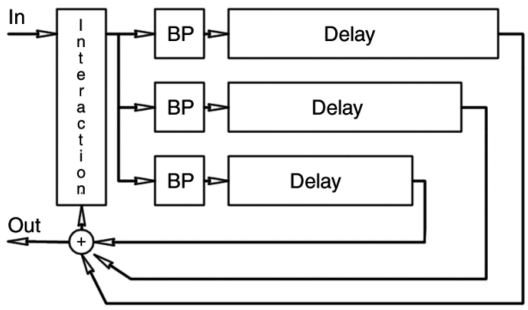

Figure 1. The banded waveguide structure

2.3. Topology

Interestingly enough there is basic structure that all waveguide and banded waveguide simulations share, independent of the specific details of the implementation. All these simulation have a basic loop connectivity and these loops are loaded at certain points for excitation. The specifics define how these disturbance then propagate, while not changing the fact that the propagation remains confined to the loop. Hence a common feature can be seen as the connectivity of the loop. One can then consider studying loop spaces directly and try to figure out properties that in fact hold for all loop configurations.

The study of connectivity is called topology. Basic operations like stretching or shrinking do not change the topology, which would correspond to a chance in length in the geometry or a change in propagation speed, while certain changes in the connectivity of the geometry do change. My interest here was to come up with a way to encode excitations topologically so that the generic features of excitations on loop spaces can be found.

The first question is: How to structurally code the excitation point in the sense of the topology of loop spaces? In some sense this is immediately suggested by what happens to excitations in waveguides. At the point of excitations initial disturbances coincide at the same spot in space, a fact that does not change by change of topology and then splits to travel to different slots in space. However there are also other points in space where these disturbances meet again. These points look exactly as if there had been an excitation there. So it is rather natural to shift to the notion of studying points of coincidence of disturbances of loops. To code coincidence topologically, we glue the loop together at points of coincidence and hence arrive at something that looks like a glued knot. A basic 8 figure is an example of a loop glued in the middle. The loops form the trajectories of the dynamics. Hence one can construct the basic glued knot types that one can observe from the dynamical situation and show that changing the dynamical properties does or does not change the knot type. It turns out that in 1-dimensional cases, which would cover strings, marimba bars, tubes or objects that turn out to be in some sense "topologically" one-dimensional, there are only two basic glued knot configurations: the pure 8 shape for a topological center configuration and a sort of double 8 with two crossings for topological off-center configurations [5] (see also figure 2). The pure 8 is the singular case where the two crossings of the double 8 fall onto the same spot and hence topologically coincide in the middle.

It turns out that the situation is much more complicated in the two-dimensional plane. Small perturbations do not preserve the topological type of the coincidence glue knot.

So basically we get the basic distinction between simplicity in one and complexity in 2 dimensions purely from loop topologies. But there are even simpler and more rich results to be had with very similar thinking. For example one can also code the sign of a displacement topologically. If an impulse points upward, it has one sign and if it points downward, it has another. If now these two states are defined by a cut, with the additional rule that points where the sign flips remain uncut, one can construct glued knots that now represent orientation in connectivity. Using this construction one can derive rather intriguing results, like the existence of inertia for certain loop types as opposed to sign-neutral and hence overall non-inertial loops for others. These can then be brought back to boundary conditions that are physically seen as sign-inverters. By the change in loop period, spectral information can be deduced from these very basic topological shapes. Hence we get a good number of observable dynamical properties just from the connectivity of the problem. These properties are not dependent in their quality on the additional information needed to describe a specific situation.

2.4. Geometry

The notion of structure also play an important role in recent developments on synthesis for drums. Again the motivation comes from waveguides. Waveguides can be seen as a discrete method that maintains the structure of the continuous solution, because it in fact discretizes the continuous solution, in form of the d'Alembert solution, directly.

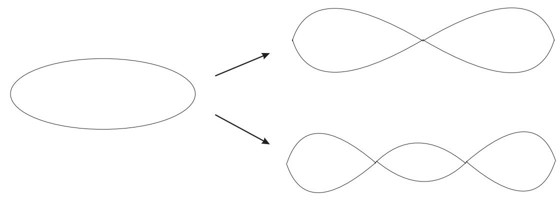

Figure 2. Two possible cases of coincidence knots from a loop related to a waveguide-like structure.

One important question here would be: What is the structure of the solution in 2 dimensions and in what sense is it comparable to the one-dimensional case. To cover this fully here would be too lengthy, and we refer the curious reader to two recent papers [6, 7]. But for this discussion I will just mention one conclusion. For one there is similar geometric structure. The waveguide can be interpreted as transporting "characteristic circles on the line under reflection" and in two dimensions the wave front is transported as "characteristic circles in the plane under reflection".

To understand the similarity, the interesting object that needs clarification is what I mean by the phrases in quotes, namely "characteristic circle in X under reflection".

The classical and informal definition of a circle is:

Definition 2.1. A circle consists of "all points at equal distance from a point in the plane."

What we mean by the above definition is really very much the same thing except that we generalize the options how two things come about or are assumed to be. These two things are "distance" and "plane".

The generalization of distance is rather intuitive. Currently I'm sitting away from a wall. If I still assume straight lines to be the shortest, then the distance from me to the wall and back well defines a distance just like the distance that we pluck into the standard definition. The difference is now that we allow for the direction in which we measure distance to abruptly change multiple times. Such an abrupt change happens for example when we hit the wall. If we used a measuring rod we would have to change the direction there. Hörmander uses the adjective "broken" for this property [10]. Now we can even generalize what we mean by going straight. If we define as straight the direction in which singularities propagate, then under inhomogeneity the direction in which a singularity will propagate may well be curved, or split or do other fun things. The propagation direction of singularities are called "characteristics". If we now connect all points of equal "broken" distance. we get a "circle" formed through these characteristics. Hence we said "characteristic circles" to indicate how these circles are formed. The side phrase "under reflection" would encode the fact that these characteristics are broken.

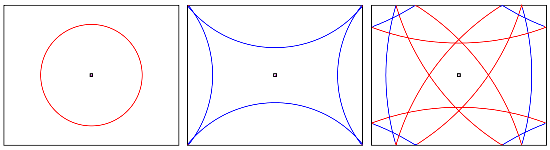

Figure 3. Reflections of a circular wavefront on a rectangular domain. The form of an ever expanding circle is maintained though segments are reflected at the wall.

Hence we have come up with a definition of a circle that is perfectly good, but generalizes to any dimensions and different geometric backgrounds (euclidean, curved, bounded...). General wave fronts on the line and in the plane nicely fit this generalized definition of the circle and this property is shared.

To see an example of these "characteristic circles under reflection" see Figure 3. Because of the boundary being straight lines, once a circle segment intersects, it folds back onto the interior of the rectangular domain, but the shape of the curve segment always stays a traditional euclidean circle segment. For general boundaries this is not the case and a circle under reflection looks locally rather different from a euclidean circle.

However, not all properties are shared between the line and the plane in this fashion. In the plane the wave equation creates solutions behind the wavefront singularity called a wake. This wake is present for solutions to the wave equation in even spatial dimensions, is simple in one spatial dimensions and is absent in all solutions of odd spatial dimensions greater or equal to 3. For some preliminary discussion of the wake in the plane in this context see [7].

3. ABSTRACT SYNTHESIS AND MATHEMATICS

So far the discussion was a review of past, sometimes recent, work in the context of physical modeling for sound synthesis. From here on we will not confine ourselves to simulating physical situations anymore, but rather look at abstract sound synthesis and we want to look into exploring mathematical structures in this context.

Basically we'll use some abstract algebra, and here predominantly very basic arrow-theoretic constructions to form on the one side a rigorous foundation to the classification of synthesis algorithm and on the other hand as a way to suggest new synthesis algorithms given the classification. This is indeed just a first step in this direction, and a general classification may have a rather different flavor once things are set and done.

3.1. Algebraisation of Sound Synthesis

What is the relationship of synthesis algorithms? By asking this question we are lead to wonder about classification. How can algorithms be distinguished or grouped together?

Classification is a task that is understood to be of interest in mathematics for a good while now. An example of the 19th century would be Klein's Erlangen Program, which basically asks for the classification of geometries by their algebraic properties.

The classification of the algebraic structure of digital signal processing algorithms, specifically Fourier transforms have also lead to structured ways of discovering new algorithms with desirable properties [14].

3.2. Algebraic Structure with Respect to Time

A first question of classification is of course, with respect to what properties we classify? Sometimes the properties are intrinsic to the objects to be classified, but sometimes this may not be the case. For this purpose assume that an audio signal essentially consists of two parts: information about time and information about the signal. Henceforth we will call T the time object and S the signal object.

The basic distinction now comes about how these two objects are generated or what their relationship is. The notation is arrow theoretic. Instead of going into detail of what it means we can read these diagrams rather casually. An arrow means, that there is a way to make what is at the arrow's tip from what is at the arrow's tail (a map, a morphism, a function, or whatever one may call this). Indicies mean different specific objects, or mappings. If there is a chain of arrows one can think of it as there being a way to get from the start of the chain to the end. Giving these basic ideas we are well equipped to look at these possible structures and give some concrete examples. At the same time we have snuck our way into basic notions of category theory [13].

3.3. Time-Signal Structures

3.3.1. Signal Driven Algorithms

First lets assume that essentially we have a way to progress between different signal objects S_i with $i \in [1 \ldots n]$. Additionally we'll know a way to create related time objects T_i using some method π_i. So we can think of this as a situation where the signal objects drive the time behavior. For this reason we could call this "signal driven". Alternatively because time is not explicitly used to drive the signal, this is a "time implicit" situation.

$$\begin{array}{ccccccccc} S_1 & \xrightarrow{g_1} & S_2 & \xrightarrow{g_2} & S_3 & & \xrightarrow{g_n} & S_n \\ \downarrow{\scriptstyle \pi_1} & & \downarrow{\scriptstyle \pi_2} & & \downarrow{\scriptstyle \pi_3} & \cdots & & \downarrow{\scriptstyle \pi_n} \\ T_1 & & T_2 & & T_3 & & & T_n \end{array} \quad (2)$$

Probably the simplest meaningful example written in functional form would look like this:

$$t = |y| + \epsilon \quad (3)$$

Time is directly derived from a signal level. We only enforce that time is strictly positive by taking the absolute value of the signal and adding a non-zero positive term ϵ. This basic example is depicted in Figure 4.

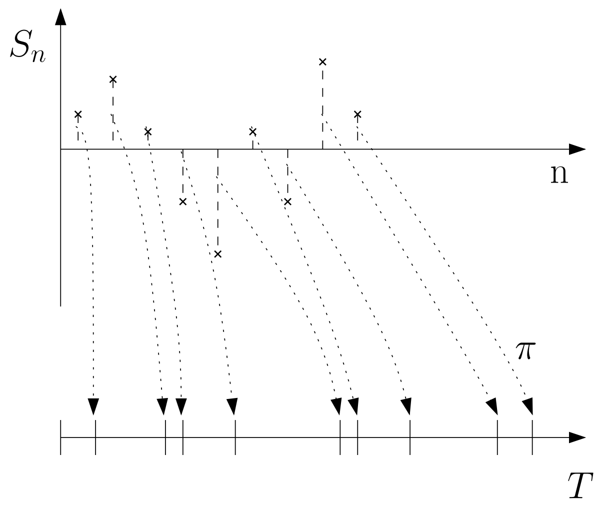

Figure 4. A basic signal driven example, where the absolute value of the signal determines the time interval between samples.

A more general version of the same form would be:

$$t = |f(y)| + \epsilon \tag{4}$$

Of course y can be something else besides the traditional function on the real numbers $\mathbb{R}$ a discrete version thereof. It can be any manifold of any dimensions, given that a mapping can be defined to bring it down to time.

A familiar example of a signal driven effect would be silence removal, where the presence of silent passages reparametrize time.

3.3.2. *Time Driven Algorithms*

We can invert the situation and assume time objects T_i and a known relation between them and derive the signal S_i from the time objects. In this case the signal is "time driven" or we have a situation where time is explicit.

$$\begin{array}{ccccccc} S_1 & & S_2 & & S_3 & & S_n \\ \uparrow \Pi_1 & & \uparrow \Pi_2 & & \uparrow \Pi_3 & \cdots & \uparrow \Pi_n \\ T_1 & \xrightarrow{f_1} & T_2 & \xrightarrow{f_2} & T_3 & \xrightarrow{f_n} & T_n \end{array} \tag{5}$$

It turns out that many classical synthesis algorithms are time driven.

Classical examples include Additive Synthesis [15]:

$$y = \sum_n A_n e^{i\omega_n t}, \tag{6}$$

all sorts of varieties of modular synthesis, like AM or FM:

$$y = A_n(t) e^{i\omega_n(t)t}, \tag{7}$$

and wavetable synthesis, where $A(t)$ is the wavetable lookup function:

$$y = A(t). \tag{8}$$

Coincidentally, this symbolically looks like the general form of this mapping written as function. The signal y is any signal that can be created from time t via any arbitrary function.

What is common for all these is that the signal is directly generated from time , i.e. time is a parameter to the function. A most confining definition could include requiring that it is the only variable parameter.

This priority to time in many classical algorithms may not be so surprising if one thinks of sound as a time series. Of course many traditional methods use one sampling rate throughout. In this case the time progression mappings $f_1, f_2, \ldots, f_n$ could be replaced by a single mapping f. The time progression maps also may depend on the current form of signal representation. For example if the signal is short-time Fourier transformed, the time progression then is defined to be the time step between Fourier transform blocks. This temporal adaptivity with respect to representation has recently been explored in the context of audio data flow by the name of "implicit patching" [2].

The idea of generalized abstraction of the time-relation goes back to a classic article by Dannenberg [4] which gives many more detailed examples of the variation one can introduce by discretely or continuously altering the time progression. A detailed discussion of time mappings with many examples can be found in [9].

3.3.3. *State Driven Algorithms*

Finally one can envision a joint object from which both time and signal are derived. I abuse notation [1] here by writing this as a product $S_i \times T_i$ of time T_i and signal S_i.

Explicit Signal and Time or State Driven:

$$\begin{array}{ccccc} S_1 & & S_2 & & S_n \\ \uparrow \Pi_1 & & \uparrow \Pi_2 & & \uparrow \Pi_n \\ S_1 \times T_1 & \xrightarrow{h_1} & S_2 \times T_2 & \cdots \xrightarrow{h_n} & S_n \times T_n \\ \downarrow \pi_1 & & \downarrow \pi_2 & & \downarrow \pi_n \\ T_1 & & T_2 & & T_n \end{array} \tag{9}$$

A basic state driven example are filters. Filters can be written as operators that act on the state by convolution. Hence if a configuration of filters H_i is applied at an instance i to a state S_i we get the new state S_{i+1} using the basic relationship:

$$S_i \xrightarrow{H_i} S_{i+1} \tag{10}$$

[1] The abuse lies with the possibility that this joint object is in fact not a product of time and signal.

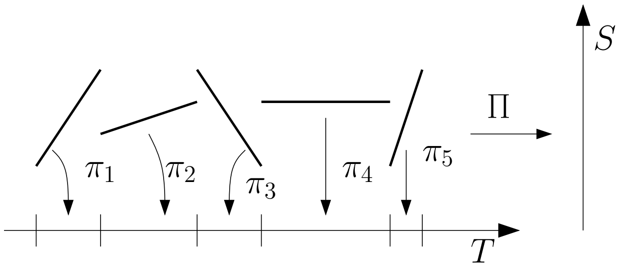

Figure 5. Example of a state driven algorithm using a rotating line projected onto the signal and the time axis.

or written as product and for all instances i [13]:

$$H \times S \to S \tag{11}$$

At first glace this notation suggests that this is in fact a signal driven method, but time was implicitly used as well. This is most obvious when the filter H itself is time-varying. To highlight this we can explicitly notate this and get:

$$S \times T \xrightarrow{H} S \times T \tag{12}$$
$$H \times S \times T \to S \times T \tag{13}$$

Lets take an example that makes the generation of time by mapping rather more obvious. Consider the following geometric state driven synthesis example. A line is rotated in the plane. The signal is generated by projecting that line in one direction and time is generated by projecting the same line in another (here orthogonal) direction and making sure that the result of the projection is strictly positive (see also Figure 5):

$$t = r|\cos\alpha| + \epsilon \tag{14}$$
$$y = r \sin\alpha \tag{15}$$

Generally any object P for which a mapping $P \xrightarrow{\Pi} S$ exists can be used to construct signal or state driven synthesis algorithms. If we confine ourselves to geometric objects, P can be any geometry of any dimensions for which we can write down a projection to a range on the real line, which is S.

3.4. Relations of Modes of Generation

Given this separation of structures one may wonder as to the relationship. For example, when can we find a way to make time explicit? Before we go to look at this question let's give an easy example where the classification above is not unique.

3.4.1. Ambiguity of Reference

We will call the possibility to classify an algorithm into more than one of the above listed types as ambiguity of reference. Let me explain this using the example of waveforms of temporal logic taken by Kauffman [11]:

$$J(t) = TFTFTFTFTFTFTFTF\ldots \tag{16}$$
$$\neg J(t) = FTFTFTFTFTFTFTFT\ldots \tag{17}$$

We have a logic sequence J parametrized by t which alternates truth value. Then we use a function $\neg$ to calculate a equally parametrized sequence $\neg J$. This new sequence has two possible interpretation as to what happened. Classically one would use what I'd call the spatial interpretation:

$$J(t) \xrightarrow{\neg} \neg J(t) \tag{18}$$

Hence the two objects J and $\neg J$ relate to each other through the operation $\neg$ and the parametrization remains unchanged.

However, a second interpretation is possible. $\neg J$ looks like J with a shift in parametrization. Hence the sequence can be interpreted temporally and the operation would be:

$$J(t) \xrightarrow{\text{shift}} \neg J(t) \tag{19}$$

The choice of reference is ambiguous hence allowing for both interpretations. In our nomenclature the first case would be signal driven whereas the second is time driven. We can meet this ambiguity also when thinking about filters.

3.4.2. Making Time or Signal Explicit

In order to be able to make time explicit in a signal driven situation, or alternatively give an explicit signal relation in a time driven algorithm one tries to find a way to come up with a mapping from $T_i \to T_{i+1}$, hence one tries to be able to find a way to draw the diagrams as follows for signal driven synthesis of equation (2):

$$\begin{array}{ccccccc} S_1 & \xrightarrow{g_1} & S_2 & \xrightarrow{g_2} & S_3 & \xrightarrow{g_n} & S_n \\ \downarrow{\scriptstyle \pi_1} & & \downarrow{\scriptstyle \pi_2} & & \downarrow{\scriptstyle \pi_3} \; \cdots & & \downarrow{\scriptstyle \pi_n} \\ T_1 & \xrightarrow{f_1} & T_2 & \xrightarrow{f_2} & T_3 & \xrightarrow{f_n} & T_n \end{array} \tag{20}$$

This is obviously possible if the mappings π_i is invertible. If this is the case then one can use the compositions $\pi_i \circ f_i \circ \pi_{i+1}^{-1} = g_i$ as equivalent progressing in time and π_{i+1}^{-1} as the function which derives the signal from time.

Essentially the same argument and steps also work for deriving a signal driven approach from time driven synthesis of equation (5). In this case we complete the diagram as follows:

$$\begin{array}{ccccccc} S_1 & \xrightarrow{g_1} & S_2 & \xrightarrow{g_2} & S_3 & \xrightarrow{g_n} & S_n \\ \uparrow{\scriptstyle \Pi_1} & & \uparrow{\scriptstyle \Pi_2} & & \uparrow{\scriptstyle \Pi_3} \; \cdots & & \uparrow{\scriptstyle \Pi_n} \\ T_1 & \xrightarrow{f_1} & T_2 & \xrightarrow{f_2} & T_3 & \xrightarrow{f_n} & T_n \end{array} \tag{21}$$

A full diagram that shows equivalent signal driven and time driven mappings for a state driven situation (9) would then look like this:

$$\begin{array}{ccccccc} S_1 & \xrightarrow{g_1} & S_2 & & \xrightarrow{g_n} & S_n \\ \uparrow{\scriptstyle \Pi_1} & & \uparrow{\scriptstyle \Pi_2} & & & \uparrow{\scriptstyle \Pi_n} \\ S_1 \times T_1 & \xrightarrow{h_1} & S_2 \times T_2 & \cdots & \xrightarrow{h_n} & S_n \times T_n \\ \downarrow{\scriptstyle \pi_1} & & \downarrow{\scriptstyle \pi_2} & & & \downarrow{\scriptstyle \pi_n} \\ T_1 & \xrightarrow{f_1} & T_2 & & \xrightarrow{f_n} & T_n \end{array} \tag{22}$$

In all these cases the crucial restriction is the existence of a unique inverse to the mappings π and Π.

4. CONCLUSIONS

The structure of a problem can help us find ways to solve it, or discover aspects that are yet unexplored. We surveyed how the idea of mathematical structures as been employed in the context of physical modeling of musical instruments. One can focus on different aspects of the problem, be it computational efficiency, topology or geometry and learn new ways of doing things by studying these aspects structurally.

Based on this idea, a few steps towards a possible algebraisation of abstract sound synthesis were proposed and we gave a few simple example of a first classification of synthesis methods with respect to the generation of time evolution. We hope that these example do illustrate that it is helpful to consider the abstract relation of algorithms and still be able to derive concrete new instances of algorithms from it.

Acknowledgment

I would like to thank Domenico Vicinanza for the kind invitation to present these ideas.

5. REFERENCES

[1] H.-W. Alten, A. Djafari-Nainim, M. Folkerts, H. Schlosser, K.-H. Schlote, and H. Wußing. *4000 Jahre Algebra*. Springer, 2003.

[2] S. Bray and G. Tzanetakis. Implicit Patching for Dataflow-Based Audio Analysis and Synthesis. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 13–16, Barcelona, Spain, September 5-9 2005.

[3] P. R. Cook. *Real Sound Synthesis for Interactive Applications*. A K Peters, Ltd., July 2002.

[4] R. B. Dannenberg. Abstract Time Warping of Compound Events and Signals. *Computer Music Journal*, 21(3):61–70, 1997.

[5] G. Essl. Aspects of the topology of interactions on loop dynamics in one and two dimensions. In U. K. Wiil, editor, *CMMR*, volume 3310 of *Lecture Notes in Computer Science*, pages 220–231. Springer, 2004.

[6] G. Essl. The Computational Structure of Waves on Drums for Sound Synthesis. In *Proceedings of Forum Acusticum*, Budapest, Hungary, Aug 29 - Sep 2 2005.

[7] G. Essl. Towards the Synthesis of Wavefront Evolution in 2-D. In *Proceedings of the International Computer Music Conference*, Barcelona, Spain, September 5-9 2005.

[8] G. Essl, S. Serafin, P. R. Cook, and J. O. Smith. Theory of Banded Waveguides. *Computer Music Journal*, 28(1):37–50, 2004.

[9] H. Honing. From time to time: The representation of timing and tempo. *Computer Music Journal*, 35(3):50–61, 2001.

[10] L. Hörmander. *The Analysis of Linear Partial Differential Operators III*, volume 274 of *Grundlehren der mathematischen Wissenschaften*. Springer, Berlin, 1985.

[11] L. H. Kauffman. Virtual Logic — The Flagg Resolution. *Cybernetics & Human Knowing*, 6(1):87–96, 1999.

[12] F. Klein. *Elementarmathematik vom höheren Standpunkt aus, Geometrie*, volume 2. Springer, 1925.

[13] F. W. Lawvere and S. H. Schanuel. *Conceptual mathematics: a first introduction to categories*. Buffalo Workshop Press, Buffalo, NY, USA, 1991.

[14] M. Püschel and J. M. F. Moura. The algebraic approach to the discrete cosine and sine transforms and their fast algorithms. *SIAM J. Comput.*, 32(5):1280–1316 (electronic), 2003.

[15] C. Roads. *The Computer Music Tutorial*. MIT Press, Cambridge, 1996.

[16] J. O. Smith. Digital Waveguide Modeling of Musical Instruments. Draft of unpublished online manuscript, available at `http://ccrma-www.stanford.edu/~jos/waveguide/`, 2003.