A Formal Composition System based on the Theory of Time Trees

Aluizio Arcela
Geber Ramalho
Spectral Processing Laboratory
Computer Science Department
University of Brasilia
Brasilia, DF 70910 Brazil
unb@brfapesp.bitnet

Abstract

The theory of time trees states that existing computable scores living inside the inner world of musical intervals can algorithmically be written for human and/or machine interpretation, if the corresponding physical phenomenon is modeled by a suitable composition of periodic motions. Such being the case, as the trajectory is going on, the time hierarchy resulting from the unequal space quantization gives rise to concurrent and mathematically inseparable rhythmic patterns and polyhedral structures. Because these geometric units are open 3D forms, a feasible strategy for the automatic construction of expressive musical pieces with time-tree primitives employs a set of functions for joining complementary parts together in the space, looking for a closure in the form. For this purpose, we describe an axiomatic formal system capable of producing spectral-event theorems (to be mapped into polyphonic scores), which is organized much as a biological circumstance, for it works by first taking a selected time-tree as a seed, that is, something carrying a music genetic code, then placing it inside a culture-medium embodied by a population of context-meaningful trees, and finally leaving them at large for spontaneous growth.

Introduction

Seven years old children are capable of computing interesting structures of numbers by the following procedure: “Take two whole numbers \( m, n \) having no common divisors, unless unity, such that \( m \) is smaller than \( n \). Then, construct two rows of numbers, where \( m \) is placed next to the center of the first row. At the second row you must put the value \( \text{Mod}(n,m) \) on the right of \( m \); on the left put the difference between top and right numbers. From this triad you must generate new rows of numbers until you find a row having \( m \) numbers all equal to \( 1 \). Now proceed in this way: every number in a row must be copied to the next row, except for one and only one (\( g \)), namely the greatest and oldest one. This selected element rather than copied must be split into a pair of descendants. One of them has to be equal to the least and oldest number (\( u \) occurring in the same row as \( g \). The value of \( u \) will be a right descendant of \( g \) if \( u \) belongs to the main right subtree, otherwise it will be a left descendant. The other descendant having in any case the value \( g-u \) [2].

As used above, the term oldest means that every number has an age-property playing a very important role in the algorithm. It basically consists of counting the number of times a number is repeated without branching. With the pair \( 8:21 \) one can see at ow the existence of two greatest values, but the first number \( 3 \) is the actual g-element, because it is two years old, while the second \( 3 \) is only one.Obviously, the number 2, being the single smallest one, is the corresponding u-element, and once \( u \) is a member of the right subtree, its value must be a right descendant of \( g \).

This kind of tree, which is called primordial tree or \( \pi \)-tree, is found by simulating the interaction of two pure tones as a system of two clocks, such that each one is trying to measure the other one. We are talking about the intervalar object - a model for just intervals which reveals the way the time is organized in the heart of the most primitive form of perceptive sounds. In fact, intervals may be seen as forests of times (durations) whose elementary parts are made from \( \pi \)-trees [2,3].

Proceedings of the International Music Conference
Reading Music from Intervals

The idea of make use of time-trees in composition comes naturally because values of frequency and intensity can be joined to the hierarchical patterns of time, so as to produce sequences of notes. A good procedure for computing these values requires an entity called Measuring Harmonic Motion (MHM), which works as follows:

Being the intervalar object the resulting \(xyz\)-space interaction between a circular motion (masked sound) and a simple harmonic motion (pitch sound), as soon as a moment \([3]\) is identified by the time-tree working, its coordinates and velocity must be read at once in order to set up a dummy periodic harmonic motion attached to the \(z\)-axis and centered on the \(xyz\)-origin.

This MHM has to have amplitude and frequency (and also the initial phase angle) strictly according to the instantaneous physical conditions of the composite motion, and such MHM parametric values are to be held for the whole time defined by the current time-tree node. Basically, the MHM measures the note pitch by reading the angular frequency of a circular motion having a radius equal to the distance from the origin to the projection of the moment over a plane, and a speed equal to the modulus of the projection of the moment velocity over a normal to the radius. This same situation is used to find the amplitude and the initial phase angle. Subsequently, the next moment actuated by the chronology of the tree will determine a new amplitude-frequency pair to be joined to the new time information, and so on, until one full period is completed. As a matter of fact, the measuring is done twice: the first one taking the intervalar object projection over the \(xz\)-plane, and the second one over the \(yz\)-plane. Consequently, a note must be played simultaneously by two instruments, or, in other words, by an ortho-stereophonic instrument: a musical device where a part of the spectrum goes to one of the audio channels, while the remainder goes to the other one \([4]\). In a certain sense, it is possible to say that a time tree is a complete score because to every note an ortho-stereophonic instrument is computed at the level of spectral components and the corresponding amplitude envelopes. To understand how this operation is done, a time-tree node must be seen as an oscillator which is a son of another oscillator, and also the father of others. Therefore, the definition of instruments is topology-dependent, for at each subtree the fundamental corresponds to the frequency of the root oscillator while the partials correspond to the frequencies of descendent oscillators. The algebraic sum of complementary and/or superimposed amplitude segments related to all descendants gives rise to a well-defined envelope for the current root.

Although infinite melodies can be found inside intervals, there are only five rhythmic patterns in the major third 4.5, as one can see above through the melodies M1 to M5. According to the theory, the object \(S:s\) has \(s\) time patterns very similar in case of simple intervals, but they are progressively...
complex with respect to similarities for large intervals (those where the quotient \( s/S \) has a long periodic decimal \([7]\)). Since the hearing of melodies M1 to M5 gives the sensation of incompleteness, and also because they correspond only to the \( xz \)-projection of the object 4:5, they are called here **incomplete melodies**.

**The TM Composition Machine**

**Moments** being temporal and spatially connected by trees are able to create a time relationship system and also to provide geometric edges among related notes (they all have well defined \( xyz \) coordinates). As from a time-tree data structure (in opposition to a time-graph) only polyhedral slices can be formed, we are led to think about that incompleteness sensation we felt on M1-M5 melodies. Would it be a symptom announcing that exists a formal grammar for melody writing which is capable of joining incomplete melodies from different intervalar objects?

To answer this question we will describe a simple machine which performs **tree adjusting** , namely a sequence of translation, rotation, and scaling, in order to achieve complementary polyhedral matching (or **tree matching**, TM for short). Some introductory definitions are listed below:

**Free Vertex**  A terminal note, that is, a moment without any descendant. It’s able to match.

**Half Vertex**  A note having only one descendant. It is also able to match.

**Full Vertex**  A note having at least two descendants. It can not match.

**Tree Matching**  The operation of put on the same coordinates free and half vertices belonging to different time-trees by adjusting one of the trees.

**Vertex Finishing**  The passage of a vertex from free to full, or from half to full.

As a metaphorical way to describe TM context, we first consider the starting time-tree as a seed, or something carrying an information package to be checked for matching with something else, more specifically with the members of a population of trees, all of them coming from intervalar objects having well defined lifetime but undefined size and \( xyz \) placement. By the way, we assume some concepts from the theory of formal systems \([6,8]\), mainly in what concerns the arrangement a theorem is built. But incompatibilities are unavoidable, due above all to the fact that to any possible TM **alphabet** each single element is computable, therefore it could not be only an arbitrary set of symbols, for each TM symbol is a data structure representation of a physical phenomenon.

**Precept** A time-tree whose intervalar object has well defined size and lifetime, and is centered on \( x,y,z \) coordinates at \( \rho_x, \rho_y, \rho_z \) rotation angles.

**Theorem**  A musically interpretable sequence of precepts.

**Seed**  The start precept, or an axiom. The seed is an unadjustable tree.

**Culture Medium**  A set of time trees having some topological relationship \([1,5]\) to the seed. Normally the medium is a meta-time-tree, i.e., a tree having nothing else than complete time-trees inside its nodes, as opposed to the common sigle duration value. As a result, the medium has hierarchical properties which allow chronologically ordered search for tree matching. All trees in the medium are adjustable.

**Growth**  A successful tree matching, or a tree matching wherein at least a certain number of **vertex finishing** occurs. The first attempt is try to match the seed to the oldest medium tree. In case of a match...
of a match success, the resulting aggregation will be the initial form, otherwise the seed will be faced to the next aged medium tree, and so on, until a stop condition is attained. According to the genetic convergence between seed and medium, this process can lead to an association of many trees or to no growth at all.

**Form** The current growth status. A theorem.

**Closed Form** A situation having no free or half vertices. It constitutes the main goal of TM.

**Stop Condition** There are three kinds of stop condition for TM: (a) The form is closed, (b) The medium has finished, and (c) Brute force, or the number of times growth must occur.

In order to conclude by using an example, let melody M4 be a seed placed in the \(xyz\) space as shown in the labeled figure on prior page. Now supposes that M4 is faced to X1, a medium element coming from the major ninth 4:9 also in the low state \([3]\). In this situation there is a lifetime condition wherein M4 is born at the right moment the object X1 is dead, such that the tree matching by adjusting X1 generates a longest melody, as shown in the melody growth figure, but yet incomplete as one can see through the number of remaining free vertices in the top figure.

**References**


I offer a formal ontological theory where the basic building blocks of the world are timeless events. The composition of events results in processes. Spacetime emerges as the system of all events. Things are construed as bundles of processes. I maintain that such a view is in accord with General Relativity and offers interesting prospects for the foundations of classical and quantum gravity. 