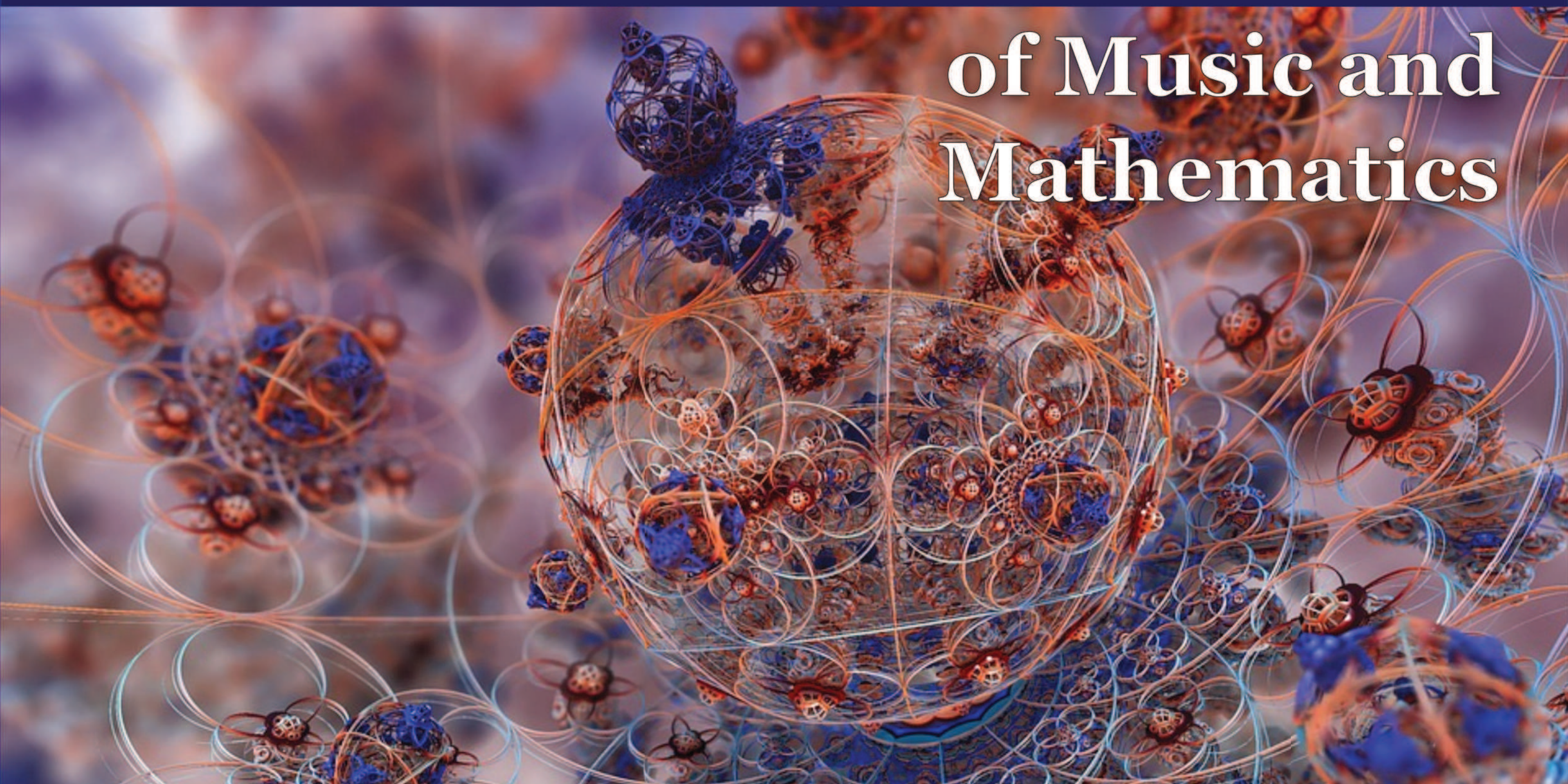
MusMat

Brazilian Journal of Music and Mathematics



Vol. III - No. 1 - Jun. 2019

MusMat

Brazilian Journal of Music and Mathematics

Vol. III - No. 1

MusMat Research Group (Ed.)

Graduate Program in Music
Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro - Junho 2019

MusMat • Brazilian Journal of Music and Mathematics

{musmat.org}

MusMat - Brazilian Journal of Music and Mathematics. MusMat Research Group (Ed.). Rio de Janeiro: Universidade Federal do Rio de Janeiro (UFRJ), School of Music, Graduate Program in Music, 2019. Vol.III, No. 1.

ISSN: 2526-3757

Contents

Foreword

The MusMat Group is very pleased to present the first number of 2019's volume of the *MusMat – Brazilian Journal of Music and Mathematics*. This issue opens with a paper by **Marianthi Bozapalidou** that introduces a *machine model* to describe fundamental music functions, such as transposition, inversion, retrogate, change of durations, pitch-class distribution, and move function. **Guerino Mazzola** proposes a mathematical construction of musical time, derived from mathematical *gesture theory* and examines its application to free jazz. **Daniel Moreira de Sousa** formalizes the concept of *textural spaces*, discussing aspects on their compositional implementation through the *modes of textural realization*. **Rael Bertarelli Gimenes Toffolo** presents a computational implementation of Pousseur's *harmonic network* in SuperCollider computer language. **Michael Winter'**s paper shows how James Tenney's *theory of harmonic distance in harmonic space* were envisioned by Leibniz more than 300 hundred years ago.

Liduino Pitombeira
Junho 2019

Machine Representation of Fundamental Musical Functions

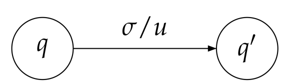
Marianthi Bozapalidou
bozapalidou@gmail.com

Abstract: *We consider a sequential machine model with output a cancellative monoid in order to describe fundamental music functions (transposition, inversion, retrogate, change of durations, pitch class distribution, move function). The minimal such machine of a prefix preserving function is provided. Musical functions are classified according the complexity of the minimal sequential transducers representing them. Functions coming from contour situations are shown to be sequential and their minimal machines are constructed. A machine simulation based hierarchy of musical contours and the corresponding classification of musical languages are exhibited.*

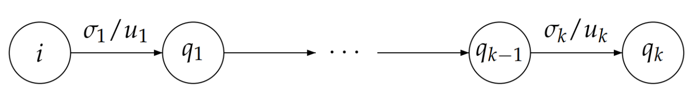
Keywords: *Musical morphism. Musical Contour. Hierarchy of Musical Objects. Sequential Transducer.*

I. Introduction

Mathematical machine models, such as automata, were already used to analyse, interpret and represent musical processes, [1], [5, 6, 7], [4], [17], [2]. Sequential transducers constitute the most general algorithm that can be executed in real time by a finite device. Non-deterministic transducers and weighted transducers have already been used in speech recognition [14, 15], natural language processing [13], [19], image generation [9] and music identification [21], [16]. A sequential transducer is a *deterministic automaton* with transitions labelled with both input and output symbols



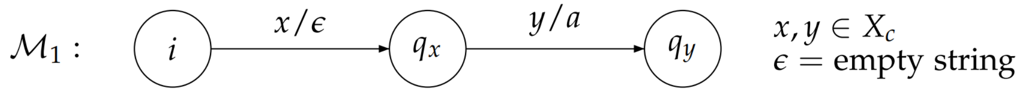
interpreted as follows: if $\mathcal{M}$ is in state q and we input the symbol σ, then $\mathcal{M}$ goes to state q' and outputs the string u. The *behavior* of $\mathcal{M}$ is the function defined in the following way: every input string $\sigma_1 \cdots \sigma_k$, labels a unique path



and the emitted string is $u_1 \cdots u_k$, where ⓘ denotes the single initial state.

Functions computed by such systems are called *sequential* and have the fundamental property to preserve *prefixes*. The preservation of prefixes musically refers to the maintenance of similarities, necessary for outlining the dynamics of the *musical flow*, thus rendering sequential transducers a considerable tool to classify musical strings.

Example I.1. Consider the chromatic alphabet $X_c = \{c, c\sharp, d, d\sharp, e, f, f\sharp, g, g\sharp, a, a\sharp, b\}$ and the sequential transducer



where $a = 1, 0, -1$ whenever y is located in X_c upwards, at the same level or downwards of x respectively. The function f_1 computed by $\mathcal{M}_1$ sends every musical string to its outline. For instance, the musical string

$$w = b\ c\sharp\ e\ b\ c\sharp\ g\sharp\ f\sharp\ e\ g\sharp\ c\sharp\ b$$

resulting from the following melodic line

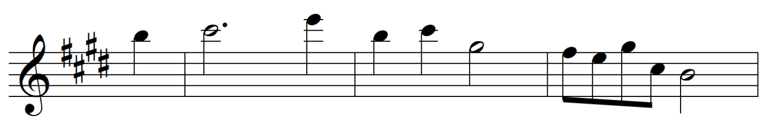
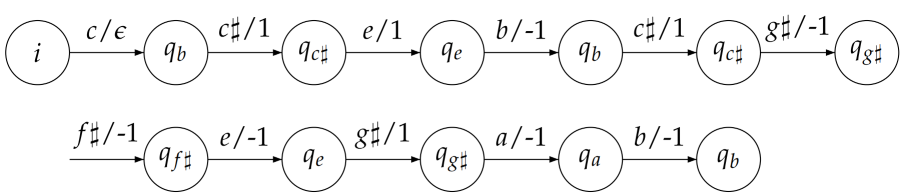


Figure 1: *Opening melodic line of Solon Michaelides' Sappho's Lyre.*

generates the path



and so

$$f_1(w) = 1\ 1\ (-1)\ 1\ (-1)\ (-1)\ (-1)\ 1\ (-1)\ (-1),$$

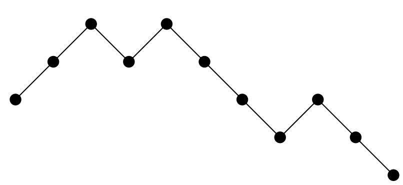


Figure 2: *Outline $f_1(w)$.*

The equivalence relation induced by f_1 classifies two musical strings to the same class if and only if they have the same outline.

The fundamental counterpoint transformations T_t (transposition), I_t (inversion) are sequential but the retrograde function

$$\sigma_1 \cdots \sigma_k \overset{R}{\mapsto} \sigma_k \cdots \sigma_1$$

fails to be sequential, because it does not preserve prefixes. On the other hand, the function counting the numbers of 0's, 1's, ... 11's occuring on a string of pitch classes, as well as the function counting the ascending, horizontal, descending moves on a musical string are not sequential because their destination sets are not free monoids. The same holds for the *move index* function f_0 that informs whether the number of ascending moves in a given musical string is greater than the number of descending moves and vice versa. In order to capture these exceptions as well, we *extend the sequential transducer model* by setting the output to be a cancellative monoid.

Another advantage of this extension is that the minimal sequential transducer $\mathcal{M}_f$ describing a given prefix preserving function $f : \Sigma^* \to M$ (M cancellative monoid) can be effectively constructed by using the residuals of f as in the classical case, [10], [20]. Musical transformations are classified according to the complexity of minimal sequential transducers computing them. If $\mathcal{M}_f$ and $\mathcal{M}_{f'}$ are isomorphic, then f and f' are *syntactically equivalent*, i.e. they represent the same mechanism regardless of the nature of the objects they act upon.

The present paper is divided into four sections. In section 2 we review some basic properties of the structure of cancellative monoids. In section 3 we propose the model of sequential transducer with output a cancellative monoid and construct the minimal such transducer associated with a prefix preserving function. Most of musical viewpoints can be represented by extended sequential transducers. Especially, musical morphisms have simple minimal machines and so they are located at the first level of any hierarchy of musical functions. The framework of the extended sequential transducers is highly appropriate to study musical contour functions (section 4).

A musical contour is a triple (Σ, M, c) consisting of a set Σ of musical elements, a cancellative monoid M of transformations or numbers and a function $c : \Sigma \times \Sigma \to M$ assigning an element $c(s_1, s_2)$ of M to any pair $(s_1, s_2) \in \Sigma^2$. The essence of music contour is an unfolding act of transition between one musical element and the next. This act of transition reflects the true substance of music, an art defined by movement in time, on the stave and the connection between theoretical/analytical significations. [2] studied contours of the form $\Sigma \times \Sigma \to \mathbb{R}$ counting quantitative features of musical strings. We show that any musical contour function with values in a cancellative monoid is sequential and we construct its minimal sequential transducer. Hierarchies of musical contours with respect to transducer simulation, as well as the corresponding musical string hierarchies, are provided.

II. Cancellative Monoids

In order to increase the recognition power of ordinary sequential transducers, we use the structure of cancellative monoid. A *monoid* is a set M equipped with a binary operation

$$\odot : M \times M \to M \quad (m_1, m_2) \mapsto m_1 \odot m_2$$

which is *associative* and admits a *neutral element* $e \in M$. Given operation $\odot : M \times M \to M$, its *opposite* $\odot^{opp} : M \times M \to M$ is defined by $m_1 \odot^{opp} m_2 := m_2 \odot m_1$, for all $m_1, m_2 \in M$. If $(M, \odot, e)$ is a monoid, then $(M, \odot^{opp}, e)$ is again a monoid, called the *opposite monoid* of $(M, \odot, e)$ and is denoted by $(M, \odot, e)^{opp}$.

For a given alphabet Σ, the set Σ^* of all finite strings (finite sequences of letters of Σ) with the concatenation operation and neutral element the empty string ϵ is a monoid, the *free monoid over the alphabet* Σ. A monoid $(M, \odot, e)$ is a *group* if the following additional condition is fulfilled: for every $m \in M$ there exists $m' \in M$ so that $m \odot m' = e = m' \odot m$. The element m' is unique with this property and is called the *symmetric* of m.

The set $\mathbb{Z}_{12} = \{0, 1, 2, \ldots, 10, 11\}$ with the clock addition

$$\begin{array}{rcl} x \oplus y & = & x + y \ \text{ if } \ x + y < 12 \\ & = & x + y - 12 \ \text{ if } \ x + y \geq 12 \end{array}$$

constitutes a group, the group of modulo12 integers. Given monoids $(M, \odot, e)$ and $(M', \odot', e')$, a function $\phi : M \to M'$ verifying the laws

$$\phi(m_1 \odot m_2) = \phi(m_1) \odot' \phi(m_2) \ \ \phi(e) = e', \ \text{ for all } m_1, m_2 \in M$$

is called a *morphism* of monoids. A morphism ϕ is an *epimorphism* (*isomorphism*) whenever ϕ is a surjective (bijective) function.

A monoid $(M, \odot, e)$ is *left cancellative* if it satisfies the condition

$$m \odot m_1 = m \odot m_2 \ \text{ implies } \ m_1 = m_2, \ \text{ for all } m_1, m_2, m \in M.$$

For a left cancellative monoid $(M, \odot, e)$ and $\alpha, \beta \in M$ a *left residual* of β by a is an element $\gamma \in M$ such that $a \odot \gamma = \beta$. If $\gamma' \in M$ also satisfies the equation $\alpha \odot \gamma' = \beta$, then $\alpha \odot \gamma = \alpha \odot \gamma'$ and so by left cancellation we obtain $\gamma = \gamma'$. This unique element (if exists) is denoted $\alpha^{-1}\beta$ and is called the *left residual of* β *by* α. In the additive case we adopt the *left difference* notation $\beta - a$. The next properties are immediate:

$$\alpha^{-1}\alpha = e, \ e^{-1}\beta = \beta, \ (\alpha_1 \odot \alpha_2)^{-1}\beta = \alpha_2^{-1}(\alpha_1^{-1}\beta), \ \text{ for all } \alpha, \alpha_1, \alpha_2, \beta \in M.$$

Right cancellative monoids are defined in a dual way. A monoid is said to be *cancellative* whenever it is both left and right cancellative.

Clearly, groups and free monoids are cancellative monoids. The set $\mathbb{N}^k$ of all k-tuples of natural numbers with the pointwise addition constitutes a cancellative monoid.

Every function from a free monoid to a cancellative monoid is said to be a *viewpoint*. A viewpoint $f : \Sigma^* \to M$ is *prefix preserving* whenever for all $s, t \in \Sigma^*$ it holds

$$f(st) = f(s) \odot \alpha_{s,t}, \ \text{ for some } \alpha_{s,t} \in M \ \text{ and } \ f(\varepsilon) = e.$$

The *residual*

$$s^{-1}f : \Sigma^* \to M \ \text{ is given by } \ (s^{-1}f)(t) = f(s)^{-1}f(st), \ \text{ for all } t \in \Sigma^*.$$

III. Sequential Transducers

We propose a sequential transducer model capable to represent musical functions that ordinary sequential transducers are unable to recognize. The output of that machine is assumed to be a cancellative monoid.

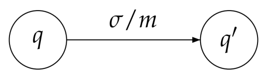
Formally, a *sequential transducer* is a *system*

$$\mathcal{M} = (\Sigma, (M, \odot, e), Q, i, K)$$

where

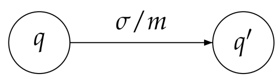
- Σ is the *input alphabet*
- $(M, \odot, e)$ is the *output cancellative monoid*
- Q is the *state set*
- $i \in Q$ is the *initial state* and

- K is a finite set of *transitions* of the form



$q, q' \in Q,\ \sigma \in \Sigma,\ m \in M$

with the property: for every state $q \in Q$ and every input letter $\sigma \in \Sigma$, there exists a unique pair $(m, q') \in M \times Q$ so that



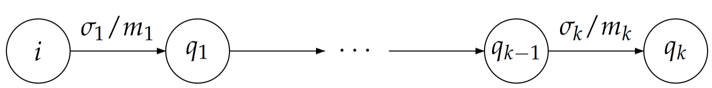
$\in K.$

The trivial transitions

$$q \xrightarrow{\varepsilon/e} q, \quad q \in Q$$

belong to K.

The *function* $f_{\mathcal{M}} : \Sigma^* \to M$ *determined by* $\mathcal{M}$ is obtained as follows: every input string $\sigma_1 \cdots \sigma_k \in \Sigma^*$ labels a unique path

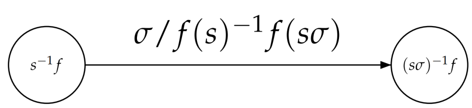


and we put $f_{\mathcal{M}}(\sigma_1 \cdots \sigma_k) = m_1 \odot \cdots \odot m_k$.

A function $f : \Sigma^* \to M$ is said to be *sequential* whenever $f = f_{\mathcal{M}}$ for some sequential transducer $\mathcal{M}$. There is a nice criterion to infer sequentiality.

Theorem III.1. *Let* $(M, \odot, e)$ *be a cancellative monoid. A prefix preserving function* $f : \Sigma^* \to M$ *is sequential, if and only if it has finitely many residuals.*

The proof follows the classical one, [10], [20]. In this case, the *minimal* sequential transducer $\mathcal{M}_f$ computing f can be effectively constructed: it has $Q_f = \{s^{-1}f \mid s \in \Sigma^*\}$ as state set, $\epsilon^{-1}f = f$ as initial state and its transitions are of the form



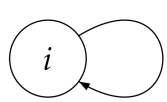
Transformations with simple minimal sequential transducers are monoid morphisms.

Clearly, every monoid morphism $h : \Sigma^* \to M$ is prefix preserving and all its left residuals coincide with h itself: $s^{-1}h = h$ for all $s \in \Sigma^*$. Indeed, for all $t \in \Sigma^*$ we have

$$(s^{-1}h)(t) = h(s)^{-1}h(st) = h(s)^{-1}(h(s) \odot h(t)) = h(t).$$

Thus, the minimal sequential transducer of h has a single state and its graph is

$\mathcal{M}_h :$



$\sigma/h(\sigma), \quad \sigma \in \Sigma.$

The transposition and inversion morphisms $T_t, I_t : \mathbb{Z}_{12}^* \to \mathbb{Z}_{12}^*$ given by $T_t(x) := x \oplus t$, $I_t(x) := -x \oplus t$ are represented by the first two graphs in Figure 3.

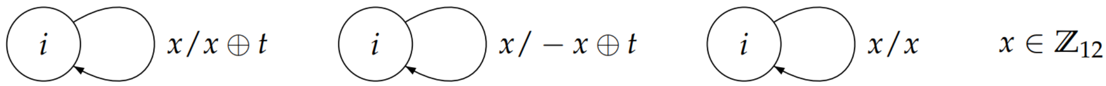


Figure 3: *The minimal sequential transducers of T_t, I_t, R.*

The *retrograde function*

$$R : \mathbb{Z}_{12}^* \to \mathbb{Z}_{12}^*, \quad R(x_k \cdots x_1) = x_1 \cdots x_k$$

satisfies the relation $R(st) = R(t)R(s)$, for all $s, t \in \mathbb{Z}_{12}^*$. Denoting by $\odot$ the opposite of the concatenation operation, $u_1 \odot u_2 = u_2 u_1$, the previous relation is written as $R(st) = R(s) \odot R(t)$, which means that R is a morphism from the monoid $\mathbb{Z}_{12}^*$ to the cancellative monoid $(\mathbb{Z}_{12}^*)^{opp}$.

Hence, its minimal sequential transducer is the third graph in Figure 3. Indeed, the output of the string $x_1 x_2 \cdots x_k$ is $x_1 \odot x_2 \odot \cdots \odot x_k = x_k \cdots x_2 x_1$ as asserted.

According to [8], a *multiple viewpoint* is a function assigning to each melody, a tuple of musical features

$$f : \Sigma^* \to \Sigma_1^* \times \cdots \times \Sigma_k^*.$$

One of the most familiar multiple viewpoints is the function $h : \mathbb{Z}_{12}^* \to \mathbb{N}^{12}$, which to every musical string $w \in \mathbb{Z}_{12}^*$ assigns the vector $(|w|_0, |w|_1, \ldots, |w|_{11})$ of numbers of $0'$s, $1'$s, …, $11'$s occurring in w. This function is a musical morphism and its minimal sequential transducer is depicted in Figure 4.

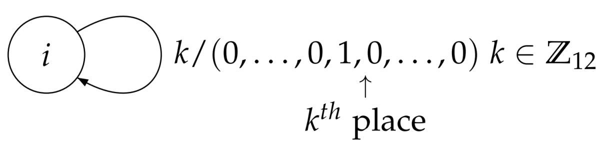


Figure 4: *The minimal machine of h.*

Another remarkable musical morphism is change in durations: we keep the same pitches and change the durations according to a function δ from a specific duration alphabet X into itself, thus imposing a new rhythm.

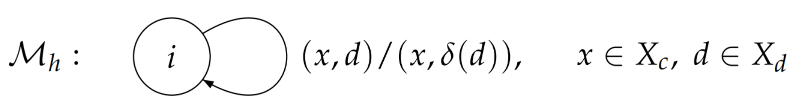


Figure 5: *Change in durations.*

The action of $\mathcal{M}_h$ according to the change of durations

$$3/4 \to 2/4, \quad 1/4 \to 1/8, \quad 2/4 \to 3/4$$

is depicted in the following example.

Figure 6: *A change of durations in Costas Nikitas' Duo for Violin and Piano*

Generally, musical morphisms are located at the first level of any hierarchy of musical functions.

IV. Musical Contours

In this section we show that any contour function with values in a cancellative monoid is sequential and we construct its minimal sequential transducer. Hierarchies with respect to transducer simulation concerning fundamental musical contour functions are provided.

Let Σ be a set of musical elements and $(M, \odot, e)$ be a cancellative monoid. Consider the contour $c : \Sigma \times \Sigma \to M$ and its associated function $f_c : \Sigma^* \to M$ defined by

- $f_c(\sigma_1\sigma_2 \cdots \sigma_k) = c(\sigma_1, \sigma_2) \odot c(\sigma_2, \sigma_3) \odot \cdots \odot c(\sigma_{k-1}, \sigma_k)$, $\sigma_i \in \Sigma$, $k \geq 2$
- $f_c(\sigma) = e = f_c(\epsilon)$, for all $\sigma \in \Sigma$ (ϵ the empty word).

Often, by abusing notation, we say that f_c itself is a contour.

The function f_c is *prefix preserving* since for all $s, t \in \Sigma^*$, it holds

$$f_c(st) = f_c(s) \odot f_c(last(s)t) \text{ and so } s^{-1}f_c = last(s)^{-1}f_c,$$

where $last(s)$ designates the rightmost letter of s. The set of all residuals of f_c is finite, $\sigma^{-1}f_c(\sigma \in \Sigma)$, $\varepsilon^{-1}f_c = f_c$ and so f_c is sequential.

Its minimal sequential transducer has states $q_\sigma(\sigma \in \Sigma)$, $q_\epsilon = i$ (initial state), where for notation simplicity we have put q_σ instead of $\sigma^{-1}f_c$. The transitions are of the form

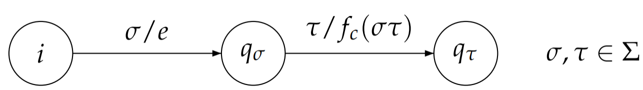


Figure 7: *Sequential transducer of a contour.*

Let X_c be the chromatic alphabet and consider the classical contour functions

$$f_1 : X_c \times X_c \to \{1, 0, -1\}^* \qquad f_2 : X_c \times X_c \to \{\ell, s, 0, -s, -\ell\}^*$$

$$f_3 : X_c \times X_c \to \{11, \ldots, 1, 0, -1, \ldots, -11\}^*$$

given by

- $f_1(x, y) = 1, 0, -1$ if y is located in X_c upwards, at the same level, downwards of x,
- $f_2(x, y) = s, -s$ (resp. $\ell, -\ell$) if y is located in X_c *one step* (resp. *more than one step*) upwards, downwards of x,
- $f_3(x, y) = k, -k$ if y is located in X_c k *semitones* upwards, downwards of x.

By applying f_2 and f_3 to the musical example of section I, we obtain

$$f_2(w) = s\ell(-\ell)s(-\ell)(-s)(-s)\ell(-\ell)(-s),$$

$$f_3(w) = 23(-5)2(-5)(-2)(-2)4(-7)(-2)$$

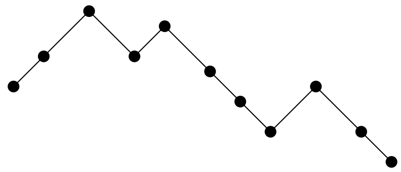


Figure 8: *Outline* $f_2(w)$.

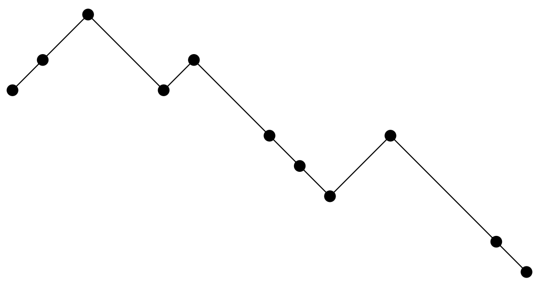


Figure 9: *Outline* $f_3(w)$.

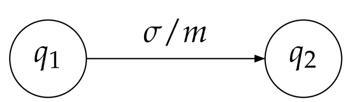
The above outlines simulate to t-norm / t-conorm-union and intersection of fuzzy (musical) sets, [3], [11].

The function $F : X_c^* \to \mathbb{N}^3$, $F(w) = (|f_1(w)|_1, |f_1(w)|_0, |f_1(w)|_{-1})$ gives information about the frequence of the ascending, horizontal, descending moves in a musical string. On the other hand the *move index* function $f_0 : X_c^* \to \mathbb{Z}$, $f_0(w) = |f_1(w)|_1 - |f_1(w)|_{-1}$ tells us whether the number of ascending moves is greater than the number of descending moves and vice versa, providing useful musical statistics for the analyst.

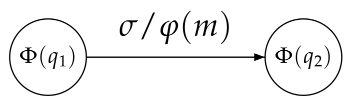
The minimal sequential transducer of f_1 was displayed in section 1. The machines $\mathcal{M}_2, \mathcal{M}_3$ of the contours f_2, f_3 are obtained from $\mathcal{M}_1$ by taking $a = \ell, s, 0, -s, -\ell$ and $a = 11, \ldots, 1, 0, -1, \ldots, -11$ respectively. Also, the machine $\mathcal{M}_F$ is obtained from $\mathcal{M}_1$ by taking $a = (1,0,0), (0,1,0), (0,0,1)$, according to $x < y$, $x = y$ or $x > y$ respectively. The above transducers are connected by simulation.

A *morphism* from $\mathcal{M} = (\Sigma, (M, \odot, e), Q, i, K)$ into $\mathcal{M}' = (\Sigma, (M', \odot', e'), Q', i', K')$ is a pair (Φ, φ) consisting of a state function $\Phi : Q \to Q'$ and a monoid morphism $\varphi : M \to M'$, so that

- $\Phi(i) = i'$ (preservation of initial states),
-

If 

is a transition in $\mathcal{M}$, then



is a transition in $\mathcal{M}'$.

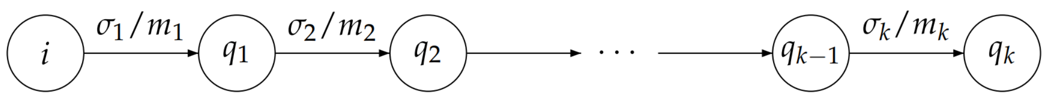
The functions computed by $\mathcal{M}$ and $\mathcal{M}'$ are strongly connected as next statement confirms.

Proposition IV.1. *Keeping the previous notation, we have*

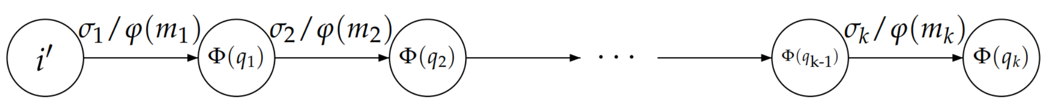
$$f_{\mathcal{M}'} = \varphi \circ f_{\mathcal{M}}$$

where $\circ$ *stands for the composition function performed from right to left.*

Proof. By definition, if



is a path in $\mathcal{M}$, then



is a path in $\mathcal{M}'$. Thus

$$\begin{aligned} f_{\mathcal{M}'}(\sigma_1\sigma_2\cdots\sigma_k) &= \varphi(m_1) \odot \varphi(m_2) \odot \cdots \odot \varphi(m_k) \\ &= \varphi(m_1 \odot m_2 \odot \cdots \odot m_k) \\ &= \varphi(f_{\mathcal{M}}(\sigma_1\sigma_2\cdots\sigma_k)) = (\varphi \circ f_{\mathcal{M}})(\sigma_1\sigma_2\cdots\sigma_k) \end{aligned}$$

hence the announced equality $f_{\mathcal{M}'} = \varphi \circ f_{\mathcal{M}}$. □

A morphism (Φ, φ) is said to be a *simulation* notation $\mathcal{M} \rhd \mathcal{M}'$, whenever both the functions Φ and φ are surjective. According to the above argument, only the states $\Phi(Q)$ and the elements of $\varphi(M)$ participate to the definition of $f_{\mathcal{M}'}$. Therefore, from machine point of view, only simulations are worthy of consideration.

Proposition IV.2. *The former contours are organized in a hierarchy*

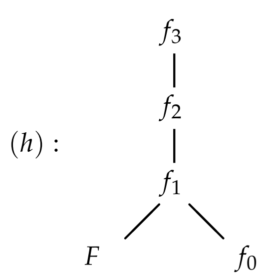


Figure 10: *Contour hierarchy.*

The notation f/f' means $\mathcal{M}_f$ simulates $\mathcal{M}_{f'}$.

Proof. First observe that all the mashines in question share a common stateset and the state functions in the simulations above are the identity functions. Furthermore, $\mathcal{M}_3 \rhd \mathcal{M}_2 \rhd \mathcal{M}_1$ via the epimorphisms

$$\phi : \{-11, \ldots, -1, 0, 1, \ldots, 11\}^* \to \{-\ell, -s, 0, s, \ell\}^*, \quad \phi(x) = \begin{cases} 0, & \text{if } x = 0 \\ \pm 1, & \text{if } x = \pm s \\ -\ell, & \text{if } x < -1 \\ \ell, & \text{if } x > 1 \end{cases}$$

and

$$\psi : \{-\ell, -s, 0, s, \ell\}^* \to \{-1, 0, 1\}^*, \;\; \psi(-\ell) = \psi(-s) = -1, \; \psi(\ell) = \psi(s) = 1, \; \psi(0) = 0.$$

Moreover, $\mathcal{M}_1 \rhd \mathcal{M}_F$ via the Parikh function $w \mapsto (|w|_{-1}, |w|_0, |w|_1)$ whereas $\mathcal{M}_1 \rhd \mathcal{M}_0$ via the epimorphism $\sigma_1 \cdots \sigma_k \mapsto \sigma_1 + \cdots + \sigma_k$.

To complete the proof we have to show that (h) is actually an ordered set. Indeed, no monoid epimorphism ϕ from the additive group $\mathbb{Z}$ of integers to the free monoid $\{-1, 0, 1\}^*$ exists, since

$$\phi(\mathbb{Z}) = \phi(\{n.(-1), n.1 \; / \; n \in \mathbb{N}\}) = \{\phi(-1)^n, \phi(1)^n \; / \; n \in \mathbb{N}\} \subsetneq \{-1, 0, 1\}^*.$$

Hence, $\mathcal{M}_0 \not\rhd \mathcal{M}_1$. Likewise, for every monoid morphism $\phi : \mathbb{N}^3 \to \{-1, 0, 1\}^*$ we obtain

$$\phi(\mathbb{N}^3) = \{\phi(1,0,0)^{n_1} \phi(0,1,0)^{n_2} \phi(0,0,1)^{n_3} \; / \; n_1, n_2, n_3 \in \mathbb{N}\} \subsetneq \{-1, 0, 1\}^*$$

and so $\mathcal{M}_F \not\rhd \mathcal{M}_1$.

The inequalities $\mathcal{M}_1 \not\rhd \mathcal{M}_2 \not\rhd \mathcal{M}_3$ come from the fact that any epimorphism of free monoids $\phi : \{x_1, \ldots, x_m\}^* \to \{y_1, \ldots, y_n\}^*$ does not increase rank, $m \geq n$. Indeed, by surjectivity, the strings $\phi(x_1), \ldots, \phi(x_m)$ generate the free monoid $\{y_1, \ldots, y_n\}^*$ and so each letter y_i is a concatenation of these strings. It turns out that $y_1 = \phi(x_{i_1}), \ldots, y_n = \phi(x_{i_n})$, with $x_{i_1}, \ldots, x_{i_n}$ pairwise distinct, i.e. $m \geq n$. $\square$

Now, let us remind some auxiliary matter. Any function $f : A \to B$ defines an equivalence w_f on the set A by setting

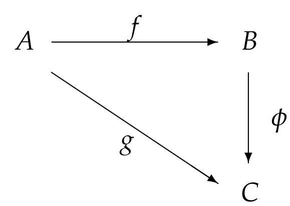
$$a_1 \equiv a_2(w_f) \iff f(a_1) = f(a_2).$$

Given equivalences w, w' on a set A we say that w is *thinner* than w', $w \setminus w'$, whenever

$$a_1 \equiv a_2(w) \;\text{ implies }\; a_1 \equiv a_2(w').$$

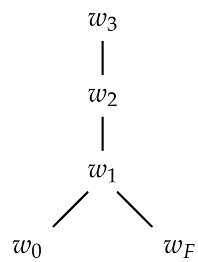
This means that any class of w' is a union of classes of w.

Consider the commutative triangle



$$g = \phi \circ f.$$

If f is a surjection, then $w_f \setminus w_g$. Putting together propositions IV.1, IV.2 and the previous discussion we get the following classification of equivalences on the set X_C^*



where w_i, w_F are the Kernel equivalences of the functions f_i, F.

V. Conclusion

Fundamental musical functions are encoded in sequential transducers. The encoding is realized by assigning to any musical function its minimal sequential transducer. A machine simulation based hierarchy of musical contours and the corresponding classifications of musical languages are provided. Our future intension will be to recognize function / transducer situations in Lewin's Generalized Interval Systems Theory, [12], as well as in self similarity theory [18].

References

[1] Barbaud, P. (1965). *Introduction á la Composition Musicale Automatique*. Paris: Dunod.

[2] Bozapalidou, M. (2013). Automata and Music Contour Functions. *Journal of Mathematics and Music*, vol. 7, No 3, pp. 195–211.

[3] Bozapalidou, M. (2018). Fuzzy Musical Languages. *Fuzzy Sets and Systems*.

[4] Chemillier, M., and Timis, D. (1990). Toward a Theory of Formal Musical Languages. *Proc. of the ICMC, Cologne*, pp. 175–183.

[5] Chemillier, M. (1990). Langages Musicaux et Automates: la Rationalité du Langage Seriel. *Actes du Colloque "Structures Musicales et Assistance Informatique"*, MIM, 36 bd Pardigon, 13004 Marseille, pp. 302–318.

[6] Chemillier, M. (2003). Synchronization of Musical words. *Theoretical Computer Science*, 310, pp. 35–60.

[7] Chemillier, M. (2004). Grammaires, Automates et Musique. *J.-P. Briot, F. Pachet (ed.), Informatique Musicale*, Traité IC2, Paris, Hermés, pp.195–230.

[8] Conklin, D., and Witten, I. (1995). Multiple Viewpoint Systems for Music Prediction. *Journal of New Music Research*, vol. 24, No. 1, pp. 51–73.

[9] Culik II, K., and Fris, I. (1995). Weighted Finite Transducer in Image Processing. *Discrete Applied Mathematics*, 58, pp. 223–237.

[10] Eilenberg, S. (1974). *Automata, Languages and Machines, vol. A*. Academic Press.

[11] Klir, G.S., and Yuan, B. (1995). *Fuzzy Sets and Fuzzy Logic Theory and Applications*. N.J.: Prentice Hall Upper Saddle River.

[12] Lewin, D. (1987). *Generalized Musical Intervals and Transformations*. New Heaven, CT and London: Yale University Press. Reprinted 2007. Oxford and New York: Oxford University Press.

[13] Mohri, M. (1997). Finite-State Transducers in Language and Speech Processing. *Computational Linguistics*, vol. 23, No. 2, pp. 269–311.

[14] Mohri, M., Pereira, F., and Riley, M. (2000). The Design Principles of a Weighted Finite-State Transducer Library. *Theoretical Computer Science*, 231, pp. 17–32.

[15] Mohri, M., Pereira F, and Riley, M. (2002). Weighted Finite-State Transducers in Speech Recognition. *Computer Speech and Language*, vol. 16, No. 1, pp. 69–88.

[16] Mohri, M., Moreno, P., and Weinstein, E. (2009). Efficient and Robust Music Identification with Weighted Finite-State Transducers. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1), pp. 197–207.

[17] Mouton, R. (1995). *Outils Intelligents pour les Musicologues*, Thése de Doctorat, Université Paris 1.

[18] Pareyon, G. (2011). Musical Self-Similarity: Intersemiosis as Synecdoche and Analogy, *The International Semiotics Institute. Imatra (ISI)*.

[19] Pereira, F., Riley, M., and Sproat, R. (1994). Weighted Rational Transductions and their Application to Human Language Processing. *ARPA Workshop on Human Language Technology*.

[20] Pin, J.-E. (2006). *A Tutorial on Sequential Functions* (available in: https://www.irif.fr/ jep/PDF/Exposes/Sequential.pdf).

[21] Weinstein, E., and Moreno, P. (2007). Music Identification with Weighted Finite-State Transducers. *Proceedings of ICASSP, Honolulu, Hawaii*.

Musical Time: A Gestural Construction

Guerino Mazzola
University of Minnesota
mazzola@umn.edu

Abstract: *We propose a mathematical construction of musical time, which is derived from mathematical gesture theory and its application to free jazz. The mathematical construction makes usage of the projective limit of diagrams of gestures.*

Keywords: *Musical Time. Gesture Theory. Improvisation. Performance.*

I. Time in Philosophy, Physics, and Music

Saint Augustin, in his confessions, states those famous words:"For what is time? Who can easily and briefly explain it? Who even in thought can comprehend it, even to the pronouncing of a word concerning it?" Time is a mysterious concept, and it has a huge impact on culture and science, see for example [4]. Philosophers have discussed time, spent many thoughts and words, but never came up with a unified understanding. Not even the time's ontological status has been clarified. Between Immanuel Kant's "Die Zeit ist kein diskursiver, oder, wie man ihn nennt, allgemeiner Begriff, sondern eine reine Form der sinnlichen Anschauung."[1] [6] and Jean Wahl's "Si le temps est qualité, il est qualité de ces événements qui se prédèdent ou succèdent ou sont contemporains les uns des autres."[2] [13, p. 308] there are fundamental discrepancies. We are not time experts in philosophy, but may refer to an excellent review [12] of those ideas, especially with regard to Paul Valéry's reflexions on time in his *Cahiers*. Let us just recall that Valéry in these writings states that time may have multiple dimensions, a thesis that has become virulent in contemporary theoretical physics in the works of Stephen Hawking [5] and Itzhak Bars [2], where time is viewed as a complex number, adding an imaginary coordinate to the usual real value.

On could argue that physics has a better concept of time since it is a basic parameter for many physical concepts: velocity, acceleration, kinetic energy, Lagrange function, etc. But beyond these basic concepts, time is also a divergent concept when one compares its role in General Relativity to that in Quantum Mechanics. In Einstein's approach, time is absorbed in a geometric space-time, which receives its curvature from the distribution of gravitational masses. In Quantum Mechanics, time is not observable. It does not correspond to a self-adjoint linear operator on Hilbert space, whose eigenvectors generate experimentally measurable quantities. Perhaps this ontological difference is one of the reasons for the present failure of a physical Theory of Everything.

In music, time seems to be a crucial variable that is at the origin of this "art of time". In this paper, we will investigate the role of time as a musical reality that differs from its philosophical

[1] Time is not a discursive or, as they say, general concept, but a pure form of sensual point of view.

[2] If time is quality, it is quality of these events that succeed or precede one another or are simultaneous.

or physical phenomenology. In fact, music shears a more constructive appoach to time. Musical time is not given *a priori*, as it is conceived in physics, but much more the result of musical creativity. The famous conductor Sergiu Celibidache, in a fascinating video about his rehearsal of Gabriel Fauré's *Requiem* with the London Symphony Orchestra [3], states that "we make time." The insight upon musical time construction could—this is our strategic hope—help generate a better understanding of time in philosophy and physics. Recall for this hope that music (and its theory) has played an important role a predecessor of other sciences. The typical example is the Pythagorean *tetractys*, which was effectively a first musical model of a physical "world formula".

i. Our Contribution to Time in Music

In this paper, we shall develop a model of musical time that is deduced from the gestural generators of musical activity. In this approach, we shall model time as a kind of "harmony among gestures", a construction that results from the intimate collaboration of musical gestures as they appear in the performative interaction of musical processes. We apply the mathematical theory of musical gestures in the framework of the category of gestures as developed in [7, 8, 11]. More precisely, we shall give a temporal interpretation of projective limits of gestures, also in view of the work of Juan Sebastian Arias [1] on topological properties of gesture spaces.

II. The Distributed Identity in Performance Is a Time Phenomenon

Our setup starts with an analysis of what was called "distributed identity" in [9]. In that model, they investigated the question of quality in free jazz performance. This is important since free jazz has no score-related abstract templates: it can only be qualified when the process of improvisation establishes a coherence of interaction in the making. Let us review this approach.

i. Distributed Identity in Free Jazz

In [9, Ch. 9.2, Ch. 11], they exhibited a phenomenon of success when the interction of musicians reached a state, where the passionate engagement is at a level, where their efforts generate a mutual understanding that flips the passionate activity into a shared stability. The music is no longer played, but plays upon the involved musicians. It was described as an axis around which the musicians rotate, becoming components of a rotational energy, which means that a higher stability of motion is achieved, this was called "distributed identity". In this model, time was not explicitely included, the model was built from an interaction of expressive gestures in performance.

ii. The Case of Score-Driven Orchestral Performance

This model seems to be limited to free jazz, but it is well known that also in traditional Western score-driven performance, the phenomenon of a distributed identity is characteristic for a successful performance. The classical example is the performance of a string quartet, which from its very beginning was conceived as a dialogue of educated persons. The interaction of voices is a basic criterion for a successful quartet performance. The interaction of voices is much more than an abstract contrapuntal architecture, it is a substantial exchange of musical gestures that transcends the score's mechanism. This is also, *mutatis mutandis*, the case in any collaborative arrangement of voices within or among instruments.

III. The Gestural Theory of a Distributed Identity

In the model of [9], the mathematical theory of musical gestures was used. We now review that approach and develop a mathematical architecture for a temporal category.

i. Morphisms of Hypergestures as Causal Units

In that model, the musicians' activities are described by hypergestures, i.e., gestures of gestures of...gestures. They are the elements of the topological hypergesture spaces $\Delta_n @ \Delta_{n-1} @ \ldots \Delta_1 @ X$, where X is a topological space (called "body") and the Δ_i are directed graphs (digraphs in short), the "skeleta" of these hypergestures. Recall from [7] that the gestures g with skeleton Δ and body X are digraph morphisms $g : \Delta \to \overrightarrow{X}$, where $\overrightarrow{X}$ is the digraph of continuous curves $c : I = [0,1] \to X$, its arrows, together with their projections to initial and terminal values as tails and heads. The set $\Delta @ \overrightarrow{X}$ of gestures from Δ to X defines a topological space, and it can be shown [1] that this topology is homeomorphic to the compact-open topology of the continuous function space $|\Delta| @ X$[3], where $|\Delta|$ is the topological space associated with digraph Δ, when its arrows are turned into real line interval I copies.

In [9], the musical dialogue between such (hyper)gestures δ_1, δ_2 is then described intuitively as "throwing a gesture δ_1 to a gesture δ_2". See Figure 1 for the example of a jazz trio. In this example, every musician's gesture lives in a space of hypergestures. Its digraphs may also be permuted in their order, and according to the Escher Theorem [7, Proposition 3.1], these permutations don't change the hyergestural spaces up to homeomorphisms. These permutations signify musically, which digraph skeleta are understood as being more on the internal or external aspect of the hypergestural constructions, see [9, Ch. 9] for the technical details behind this musical understanding of hypergestures. In Figure 1, the permutations are signaled by short red arrows, while the long arrows between different musicians signal the permutation that is selected for a throwing activity.

In [9, Ch. 9.2], the mathematical restatement of the intuitive (musi)causal "throwing gestures" from (hyper)gesture δ_1 to a (hyper)gesture δ_2 is given by a morphism of gestures. A morphism $f : \delta_1 \to \delta_2$ is defined as follows. If $\delta_1 : \Delta_1 \to \overrightarrow{X_1}, \delta_2 : \Delta_2 \to \overrightarrow{X_2}$, a morphism f is defined to be a pair $f = (h,k), h : \Delta_1 \to \Delta_2, k : X_1 \to X_2$, where h is a digraph morphism and k is a continuous map, such that the diagram

$$\begin{array}{ccc} \Delta_1 & \xrightarrow{\delta_1} & \overrightarrow{X_1} \\ h\downarrow & & \downarrow\overrightarrow{k} \\ \Delta_2 & \xrightarrow{\delta_1} & \overrightarrow{X_2} \end{array}$$

of digraphs commutes. Here, the map $\overrightarrow{k}$ is the evident digraph morphism induced by k. Such a morphism is also associated one-to-one with the commutative diagram of topological spaces

$$\begin{array}{ccc} |\Delta_1| & \xrightarrow{|\delta_1|} & X_1 \\ |h|\downarrow & & \downarrow k \\ |\Delta_2| & \xrightarrow{|\delta_1|} & X_2 \end{array}$$

Refer also to [1] for this fact.

[3] We denote by $A@B$ the set of morphisms from A to B in a determined category.

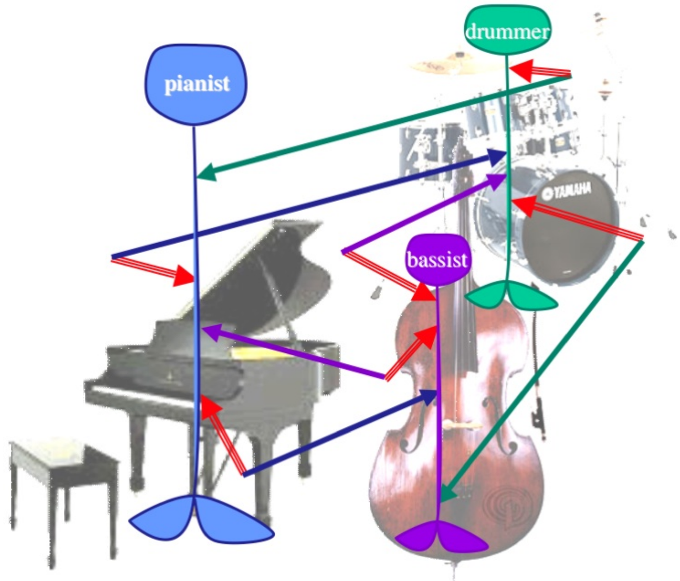


Figure 1: *The musical dialogue between such (hyper)gestures is described intuitively as "throwing gestures" from musician's to musican's gestures. Here in a jazz trio.*

ii. The Diagram of Distributed Identity

This setup reinterprets the "throwing" actions as a diagram $\mathcal{D}$ of gestures δ_i in the category *Gesture* of gestures, which define the present orchestral setup (in free jazz or elsewhere).

In this approach the morphisms $f_{i,j,k} : \delta_i \to \delta_j$ between the given gestures play the role of the musicians' understanding of their relationships to their fellow musicians. Be aware that there might be several morphisms between the same couple of gestures. There might also be endomorphisms $f_{i,i,k} : \delta_i \to \delta_i$, which describe the dialogue with oneself, which Cecil Taylor stressed in his discussion of lonely years without gigs, playing alone at home and listening to oneself. This setup is not thought to happen within physical time, it is part of the imaginry reality of the musicians' artistic presence. We don't discuss this aspect here, but refer to [10, Ch. 2.3], where a precise space of artistic presence was discussed.

IV. Time as a Projective Limit Structure

So far the idea of a distributed identity in [9] was not made explicit in mathematical terms. They establish a diagram $\mathcal{D}$ in the category *Gesture* of gestures, but the criterion of a "rotational axis" is not made precise. To put it into critical words: Why would such a diagram guarantee any success, in free jazz, say?

We now want to make this criterion more precise: What would be a precise mathematical statement of the existence of such a "rotational axis"? It must be a structure that emerges from the diagram $\mathcal{D}$, but is not automatically realized by that diagram.

i. The Projective Limit of Communicating Hypergestures

Given a diagram $\mathcal{D}$ in a category, it is in general not true that the projective limit **Limit**$\mathcal{D}$ of the diagram exists. Let us look at the situation in the category *Gesture*:

Theorem 1 *In the category Gesture, a diagram $\mathcal{D}$ has a projective limit* $\mathbf{Limit}\mathcal{D}$*. It is the projective limit of the domain skeleta, being mapped into the projective limit of the bodies' digraphs by means of the canonical morphism of projective limits. Moreover, if one views the gestures and their morphisms $\delta_i : \Delta_i \to \overrightarrow{X_i}$ within $\mathcal{D}$ as being represented by corresponding continuous maps $|\delta_i| : |\Delta_i| \to X_i$, then the projective limit is the projective limit of this diagram of continuous maps, i.e., a continuous map $\mathbf{Limit}|\Delta_i| \to \mathbf{Limit}X_i$.*

The theorem's proof is straightforward in view of the functorial correspondence between digraph and topological space representation of gestures, see also [1].

The critical point of this projective limit construction is that the limit might be empty if the "throwing morphisms" are not sufficiently compatible. For example, if two musicians interact with two morphisms $f_1 : \delta_1 \to \delta_2, f_2 : \delta_1 \to \delta_2$ such that no two skeletal points are mapped to each other, the limit will be empty. This example makes clear that the existence of points in $\mathbf{Limit}\mathcal{D}$ means that there is a mutual understanding among these musicians. Clearly, the limit will not be empty if the diagram's directed graph has no two morphisms ending on the same codomain.

A first simple example of a potentially non-empty limit is the situation, where we only have arrows from the musicians' gestures to the conductor's gesture in an orchestra. Here the morphisms ending in the conductor's gesture must end on common points of the conductor's gestural skeleton. This is also what one understands when agreeing that the orchestra follows the conductor's gestures. A similar situation is derived from musicans playing according to a metronome's gestures.

In any case, the idea of a rotational axis for a distributed identity will now be made precise in the sense that

the rotational axis of a distributed identity of a gestural diagram $\mathcal{D}$ is the set of points in $\mathbf{Limit}\mathcal{D}$. This is equivalent to having a non-empty $domain(\mathbf{Limit}\mathcal{D}) = \mathbf{Limit}\ domain(\mathcal{D})$ of the topological spaces of the diagram's skeleta. And this again is equivalent to the existence of gestures $I@\mathbf{Limit}\mathcal{D}$, or gestures $\Delta@\mathbf{Limit}\mathcal{D}$ for any skeleton Δ (empty skeleta are not allowed).

ii. Temporal Gestures in a Projective Limit

We now use the projective limit construction (well, its existence) in the category *Gesture* as a point of departure for a temporal interpretation of a distributed identity's rotational axis. What has been achieved when we have a gesture $\delta : \Delta \to domain(\mathbf{Limit}\mathcal{D})$? We have a continuous map from $|\Delta|$ to the domain limit of the diagram's skeleta qua topological spaces. This means that the connecting morphisms enable points that correspond to each other within the skeletal parameters.

This situation means that the local skeletal parameters of musicians' gestures are now connected by morphisms and generate limit points, i.e., they define a "harmony of local gestural parameters" of the distributed identity. We understand this as a construction of a "global gestural parameter", a parameter of mutual congruence in the gestural interaction. Such a global gesture is what we now define as being a musical time of $\mathcal{D}$. It means that the totality of the 'orchestra' shares gestural parameters that are exchanged via the diagram's morphisms between the musicians' gestures.

V. Conclusion

The present construction of a temporal parameter of shared gestures makes use of the projective limit of a diagram of gestures that describes the musical interaction of an orchestral setup (in any style of music, not only free jazz). This approach confirms Jean Wahl's claim that time is only generated from an environment of events, but it also uses the mental construction of a projective limit, which is not, as such, derived only from these events, and in this sense refers to an *a priori*, but not identical to Kant's approach. It is not a point of view, it is a mental—and this must be stressed: a musically conceived—construction.

References

[1] Arias, J. A. (2018). Spaces of gestures are function spaces. *Journal of Mathematics & Music*, vol. 12, no. 2, pp. 89-105.

[2] Bars, I. (2009) *Extra Dimensions in Space and Time (Multiversal Journeys)*. Heidelberg: Springer.

[3] Fauré's Requiem - Sergiu Celibidache in Rehearsal - London Symphony Orchestra (BBC 1983) (6/6). https://www.youtube.com/watch?v=f6UlqqKE6cI.

[4] Demandt, A.(2015) *Zeit - Eine Kulturgeschichte*. Berlin: Propyläen.

[5] Hawking, S. 1988 *A Brief History of Time: From the Big Bang to Black Holes.*. New York: Bantam Books.

[6] Kant, I. 1956 *Kritik der reinen Vernunft*. Hamburg: Meiner.

[7] Mazzola, G., and Andreatta, M. (2007). Diagrams, gestures and formulae in music. *Journal of Mathematics & Music*, vol. 1, no. 1, pp. 23-46.

[8] Mazzola, G. (2009). Categorical gestures, the diamond conjecture, Lewin's question, and the Hammerklavier Sonata. *Journal of Mathematics & Music*, vol. 3, no. 1, pp. 31-58.

[9] Mazzola, G., and Cherlin, P. (2009). *Flow, Gesture, and Spaces in Free Jazz*. Heidelberg: Springer.

[10] Mazzola, G., Park, J., and Thalmann, F. (2011). *Musical Creativity*. Heidelberg: Springer.

[11] Mazzola, G., et al. (2017). *The Topos of Music III: Gestures*. Heidelberg: Springer.

[12] Ustyn, B. (2018). "Time is Production": Process-Art, and Aesthetic Time in Paul Valéry's *Cahiers*. *Humanities*, 7, 4; https://doi.org/10.3390/h7010004.

[13] Wahl, J., (1986). *Traité de Métaphysique: Cours Professés en Sorbonne*. Paris: Payot.

Composing with Textures: A Proposal for Formalization of Textural Spaces

*Daniel Moreira de Sousa
Federal University of Rio de Janeiro (UFRJ)
danielspro@hotmail.com

Abstract: *This paper presents the networks of textures called* textural spaces*. Each textural space provides not only various ways of encoding the organization of the component parts of a texture, but also the quality of their relations.* The textural class space *is the most generic description of a texture as it divides the components into two abstract structures:* line *and* block*. This basic components are defined by the quality of their appearance and functionality, determined by the uniqueness versus multiplicity of the sounding components therein. The second textural space, called* ordered partition space*, consist of ascribing an integer partition to specify the number of components within a textural class. Finally, the* partition layout space *provides the most refined description of a texture among all textural space since it considers the internal order of the components according to their registral placement or timbre distribution. After presenting the various concepts and operations, the paper concludes with a discussion about the potential creative application of them, and how they can be decoded into a musical score.*

Keywords*: Musical Texture. Textural spaces. Music Composition. Music Analysis. Theory of Integer Partitions.*

I. Introduction

Music texture is one of the most important aspects of the creative process, not only for allowing the unfolding of the music form through its transformations but also for organizing the role of each component within the compositional structure. Despite its importance, there is a lack of systematic studies on texture, principally if compared to other musical parameters. In fact, many recent composers do not explicitly relate the textural organization of their music to any systematic approach. Since their music encompasses various textures, possibly, these are conceived intuitively as the outcome of the manipulation of other musical parameters.

Perhaps this is due to the diffuse and sometimes elusive definition of the term texture in music, often used rather loosely to describe both the organization of simultaneous voices or instrumental part (texture as a structure) and the overall quality of a piece defined by its instrumental techniques and combinations considering its sonic perspective (texture as sonority) [8, p. 93]. Therefore, a more precise definition shall be provided.

*This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. A special thanks to Prof. Dr. Robert D. Morris, who helped me to formalize most of these concepts during my period at Eastman School of Music (Rochester, NY).

Following Wallace Berry, by music texture we refer to both the quantitative and qualitative aspect of music [3]. As a quantitative aspect, it refers to the number of simultaneous *threads of sounds*[1]. The qualitative aspect consists of examining the relations between and among the threads of sounds to set the components of the texture (layers or strata).

An important part of the compositional process consists of organizing the textural components considering the hierarchical sense between and among them, their chronological and diachronic presentation, the way they are combined, and the like. Yet, texture is not a matter of style although various music styles can be associated with a specific texture or group of textures. A texture became recognizable by its morphology, i.e., the mutual relations of its internal components rather than their particular characteristics. Hence, a given texture can be found in different works.

Most of the studies on texture have been devoted to an analytical perspective as an alternative to the traditional methodologies based on pitches and rhythm, and their relations with the music form. Apart this, the creative use of texture is most often confined to the pre-selection of textural configurations or generalized concepts of textural behavior, without necessarily presenting any theoretical discussion that justifies such choices. Therefore, the selection results of the composer's personal experiences and creative ideas.

Another common compositional approach to texture consists of describing either the generic organization of parts through conventional labels of classification such as *monody*, *homophony*, *polyphony*, and *heterophony*, or some technical procedures involved in the compositional process that have been developed since twentieth century, such as *micropolyphony*, *pointillism*, *stratification*, and *sound-mass*. This demonstrates the lack of a clearer and more precise refinement for using texture as an autonomous element during the creative process.

In Brazil, there was an special interest in the subject, mainly in the last decades.[2] Some of these authors were concerned not only with the analytical potential of texture formalization, but also with the possible application to musical composition, especially within the scope of the MusMat research group, to which the present author belongs.

Departing from this background, this paper introduces the concept of *textural spaces*, a formalization of different networks of textures. For each textural space, a topological study is developed to examine the connections between textures in each space. This paper concludes by presenting some of the potential application of the textural spaces within the compositional process, including a brief discussion on how they can be decoded into a music score.

II. Textural Spaces

The *textural spaces* are out-of-time networks that list all possible textural structures connected or related by some operations.[3] All possible textures can be encoded within any textural spaces since they differ from one another in their level of details and by the involved operations that connect their elements, although the principle of these operations is common among them. Thus, the textural spaces are subordinated to each other. The details of each textural spaces are related to how a given texture can be evaluated either by examining the score, to observe the way its components are combined regardless their specific nature (actual pitches and durations), or by its

[1] The term "threads of sounds" (or simply threads), a musical allusion to the threads that constitute the weft of a fabric, refers to the number of concurrent sonic events that produces a linear continuity. For example, a four-part polyphony consist of four threads, and a six-note chord comprises six threads.

[2] See, for example, [16], [30], [1], [31], [15], [10], [29], [17], [4], [21], [7], [27], among others.

[3] The textural spaces are, at a certain level, based on Morris' *pitch spaces*, in which the possible relations among the various concepts of pitches are mapped into an exhausted taxonomy ([25]). This proposal is a further development of a previous work [22].

aural perception. Once we understand the aspects involved in all spaces, we are able to discuss their potential to coordinate the texture throughout the creative process.

i. textural class Space (tc-space)

A given texture can be described as the organization of simultaneous threads of sounds into *lines* (L) and *blocks* (B). Each thread is differentiated from the others by its own unique features, but if two or more threads share some characteristics, such as rhythm, note onset times, or register placement, or they have a similar timbre, melodic contour, dynamic, and so on, they can be grouped into a block. Otherwise, each thread not related by similarity with the others constitutes a line. Thus, the difference between blocks and lines is defined by their *thickness*, that is, the number of threads thereof, so that a line could be understood as a block with a single thread, and, in the same way, a block may have two or more threads to differentiate from a line[4].

Each line and block constitutes what we call *textural part* or simply *part*, a layer or strata within a musical texture (musical weft).[5] The morphology of a given texture concerns the *multiplicity* of its textural parts and its correspondent thickness. For instance, a passage of music might have three textural parts: one block and two lines, and the block is formed by four threads. Another passage also may have three textural parts, but three blocks, with two of them having three threads and the other one with five threads. The multiplicity of each textural part can be expressed by a superscript positive integer in the form of $[L^x B^y]$, where x and y are positive integers greater than or equal to zero. For example, $[L^0 B^3]$ stands for three blocks, with the same thickness or not, and $[L^2 B^0]$ indicates two lines. When the superscript is 1, it is omitted, provided that $[L^1]$ and $[B^1]$ are special cases denoted, respectively, by $[L]$ and $[B]$. Also, in order to facilitate the notation, when x and y are equal to zero, we may omit the correspondent part. Thus, $[L^0 B^3]$, for example, is rewritten as $[B^3]$ and $[L^2 B^0]$ is rewritten as $[L^2]$.

The typology of Ls and Bs consist of the most general textural description since it is grounded only in the relative distinction of the textural parts without discerning the exact thickness of the blocks. If the actual multiplicity is ignored for purposes of argumentation, we can preserve the superscript as x and y only to indicate any integer greater than or equal to 2. This general representation of texture using Ls and Bs is called *textural class*. A textural class encompasses all textures that share a similar organization of parts, ignoring the actual thickness and multiplicity thereof. Since texture is not a matter of style and does not depend on the nature of musical materials that differentiate one piece from another, the same textural class can be used in different ways. For example, all textures with a single line and two or more blocks of any thickness are members of the textural class $[LB^y]$. Altogether, there are eight possible textural classes:

1. A single line ($[L]$);
2. A single block ($[B]$);
3. A single line and a single block ($[LB]$);
4. Multiple lines ($[L^x]$);

[4] Thickness can also be defined by the span in the register considering the number of threads at intervals other than unison. Thus, a line could have more than one thread if all of them are in unison and they are perceived as a single and thin unity. In this case, the difference between lines and blocks would be related to the aural perception, taking into account the qualitative observation of the threads' organization instead of the simple calculation of their amount. We have left out this perspective to keep the discussion basic.

[5] The organization of the threads into lines and blocks shall be contextual and argumentative; each analyst or composer can determine his/her own criteria.

5. Multiple blocks ([B^y]);
6. Multiple lines and a single block ([$L^x B$]);
7. A single line and multiple blocks ([LB^y]);
8. Multiple lines and blocks ([$L^x B^y$]).

All textural classes derive from the classes L and B by combining both themselves and with each other. The set of all textural classes constitutes the simplest textural space called *textural class space* or *tc-space*. Given the large number of all possible textures, the formulation of a textural class space is useful because it defines equivalence relations to group textures with a similar morphology into a finite number of classes. Each textural class can be associated with one of the conventional labels of texture (monophony, polyphony, heterophony, and homophony), but providing a more detailed (and even clearer) description of the organization of their component parts. Indeed, the classes are apart from any aesthetical-stylistic association, which increases their creative potential in various ways.

According to the quality and the number of the parts, we can group the eight textural classes into four different types. First, considering the classes formed by only one type of textural part, we have the first two types *monopart* and *polypart*. Whereas the monopart type encompasses classes with a unique part ([L] and [B]), the polypart includes classes with a noted multiplicity ([L^x] and [B^y]). Now if we consider the combination of both types of textural part, we have the last two types: *isopart* and *heteropart*. In the isopart classes, the parts have a balanced distribution of multiplicity, that is, both or neither parts are multiple ([LB] and [$L^x B^y$]).[6] In contrast to isoparts, the multiplicity of the heteroparts is restrict to one of the parts ([$L^x B$] and [LB^y]).

All types can be related to monoparts by a given derivation process, which may allow a smooth change between them. Polypart classes constitute an expansion of monoparts to a polyphonic context, given that monoparts are included in the polyparts. Another way to think about the isopart classes is to note that [LB] and [$L^x B^y$] correspond, respectively, to the union of the two classes within monopart and polypart classes. Lastly, the unbalanced multiplicity of heteroparts outcomes from the combination of a monopart with a polypart class of a different nature (L and B).

Figure 3 summarizes the formation process of all textural classes from the combination of monoparts (ordinary circles) and polyparts (dashed circles). Each line indicates the union of classes to produce the other textural classes, also revealing the present of derivative relations between the four textural types mentioned above. Lines within the same type (dotted lines) indicate the formation of isoparts, while the cross relations (double lines) show the formation of heteroparts. The relation among the four types (and, consequently, among all textural classes), at some level, explicits the complexity hierarchy of them.[7] This hierarchy takes into account the abstract nature of the classes, in which the actual number of threads involved is undefined. Otherwise, the most complex texture of n number of threads would be [L^x] (maximum of threads and parts). Nevertheless, we are not able to determine whether a multiple part has two or more elements, nor the exact number of threads within a block. Thus, the definition of relative complexity shall consider the minimum number of parts to constitute a class, which enables [L] to be the simplest class, while [$L^x B^y$] is the most complex, given that it is formed by at least four textural parts including blocks and lines (the higher number among all classes).

[6] Obviously, this balance is not necessarily concrete since, in tc-space, the actual multiplicity is not precisely defined.

[7] Complexity here is based on the relation between the number of threads and the way they are organized. The greater the number of threads and parts, the greater the complexity. In this case, a single line is simpler then any block [21][19][18].

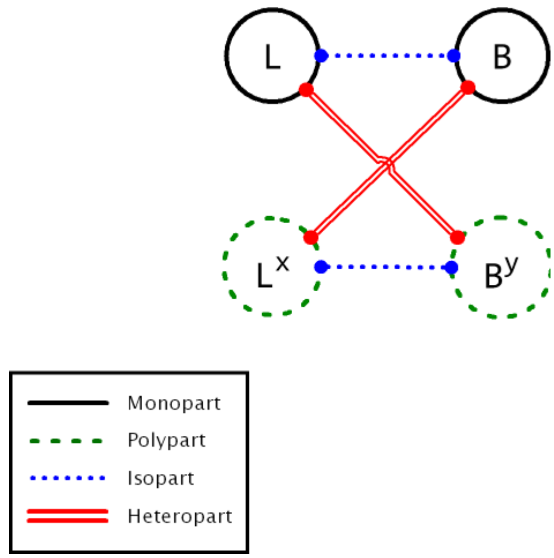


Figure 1: *Textural Classes: process of formation and the relation among the four types.* monoparts *(ordinary circles),* polyparts *(dashed circles),* isopart *(formed by the union of dotted lines), and* heteropart *(the union of double lines).*

To see how textural classes are defined within a piece, and the way they interact to each other, let us examine an excerpt of Mozart's *Eine kleine Nacthmusik*. The annotated textural classes below the score are defined by the rhythmic coincidence of the threads according to the vertical alignment thereof so that, the threads that share the same rhythmic are in collaboration to assemble a block (indicated by blue squares), and the threads with no rhythmic coincidences constitute lines (red squares - Figure 2a).[8] Each instrument of the string quartet corresponds to a single thread since there are no double stops. The excerpt is made out of only three different classes ([B], [B^y], and [LB]), that are combined to produce a textural sequence of five classes. From an imitative process, the monopart [B] is duplicated producing a polyphony of blocks (polypart class [B^y]). The ornamental variation on the rhythm in the first violin divides the first block in such a way that the first violin becomes an isolated line, while the second violin merges to the block of the viola and cello, forming the isopart class [LB]. Given that each class spans a whole measure, the brief use of class [B] within measure 13 can be understood as an deviation of [LB] (a *neighbor texture*[9], a possible clue for the block in measure 14 (or even an anticipation of it).

The graphic in Figure 2b, called *textural flow*, provides a temporal representation of this sequence to facilitate its overall observation.[10] By examining the curve in the Textural Flow, we notice that the first two classes are, respectively, the simplest (class [B]) and the most complex (class [B^y]) textures of the excerpt. Therefore, the interval between them is the greatest possible leap, but due to the melodic content of the parts in the imitative procedure, the textural gap is

[8] A given part is differentiated from the other by comparing their threads. All threads within a textural part are in *contraposition* relation with the threads of the other parts. The opposite relation of contraposition, called *collaboration*, determine whether the threads may form a block or not. Both concepts are based on Berry's independence and interdependence relations (see [3, p. 185]).

[9] For a further discussion regarding structural and ornamental functions of texture, see [23] and [21]).

[10] textural flow is a two-dimensional graphic that includes in the x-axis the duration of each class, and in the y-axis, all classes hierarchically ordered according to a relative complexity. This graphic enables the observation of both the dominance zones of a given textural class by the formation of plateaus, and the unstable or ornamental character of a class by its peaks.

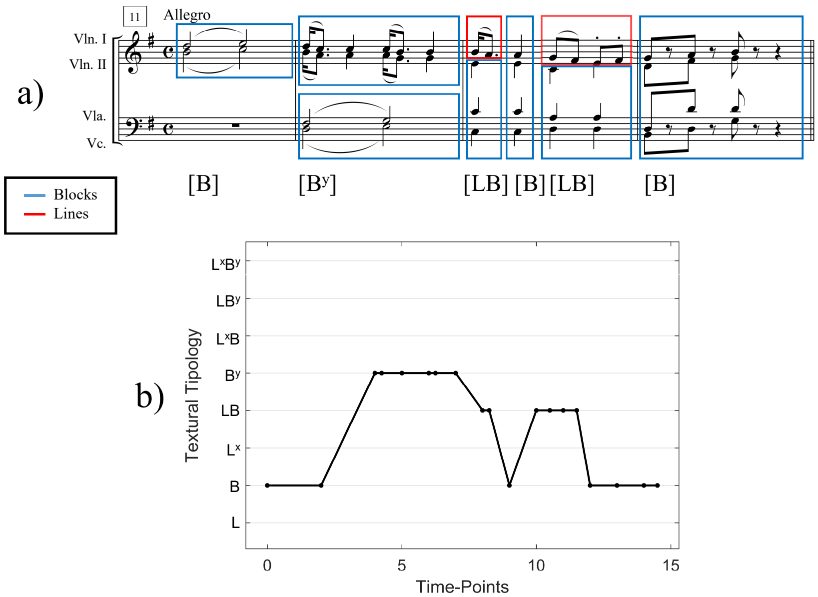


Figure 2: *Textural analysis of Mozart's* Eine kleine Nacthmusik*, K. 525, mm. 11-14 (1787): a) Annotated score with the correspondent textural class; b)Textural Flow graphic presenting the textural classes over time.*

attenuated. Moreover, although with different thickness, the blocks are associated with the same class, which implies that the overall movement of the texture describe and arch-shape trajectory departing from class $[B]$.

Operations on tc-space

Once we have defined the textural classes, we shall now discuss the relational topology of them. The inclusion of a single thread in a given texture may affect either the number of parts, or the thickness of them. This alteration depends on whether the new thread is in collaboration or in contraposition relation with the others. For example, adding a thread in textural class $[L]$ may result in $[L^x]$ or $[B]$. However, textural class $[B]$ can either turn into $[LB]$ or remain the same as a thicker block. In this case, the class would be preserved as $[B]$. Thus, the discussion of how a texture class can be related to another shall concern the changes in the *textural dimensions* (multiplicity and thickness).

Two basic operations may be defined regarding each dimension: *layering* (Y) and *shifting* (F).[11] These operations can be understood as functions for which the output is a textural class different from that of the input, but preserving some features. Therefore, the operations reveals possible adjacencies within tc-space. Both operations involve necessarily a change in the number of threads in the texture in either way increasing or decreasing it, that is, their application demands the inclusion or exclusion of threads in the textural class.

The layering operation refers to the inclusion or exclusion of a textural part, noted as $*Y_n$, where "*" is either a + or – symbol, which indicates the orientation, and n correspond to either

[11] Both operations are based on the relational topology of partitions, called *partitional operators*, proposed by Gentil-Nunes [10, 45-50].

L or B.[12] Thus, $+Y_n$ denotes the inclusion of a new block or line in the class; $-Y_n$ denotes the exclusion of a line or a block from the class.[13] For example, given the textural class K, $+Y_L(K)$ is the addition of a single line to K, while $-Y_B(K)$ removes a block from K. If K is equal to LB, then $+Y_L(K)$ is equal to $[L^xB]$, and $-Y_B(K) = [L]$. Henceforth, we shall use Y to refer to the notation $*Y_n$.

Given the relative description of the parts, in which the actual thickness is not defined, Shifting operation (F) refers to the transformation of a block into a line, and vice versa. Thus, $+F$ operating on a textural class K (that is, $+F(K)$) increases the thickness of one of any line, changing it into a block, whereas $-F(K)$ decreases the thickness of one of any blocks, turning it into a line. For instance, given the textural class $[LB]$, $+F([LB])$ indicates that the $[L]$ in the class becomes a Block, resulting in the class $[B^y]$. Similarly, $-F([B])$ is equal to $[L]$. Henceforth, in general F will denote $+F$ or $-F$.

When a part has multiplicity, operation F has more than one possible output. As the exact number of multiplicity for each part is not specified, we shall consider all possible values. For a multiplicity factor of two, the minimum necessary to form a multiple class, we have one possible output for F, whereas for any number equal to or greater than 3, F produces a different outcome. Therefore, except by $[L^xB^y]$, where one of the outputs corresponds to itself, all multiple class have two possible outputs under F. For example, $+F([L^x])$ is equal to $[LB]$, for $x = 2$, and to $[L^xB]$, for $x \geq 3$. Likewise, $-F(LB^y)$ is equal to $[L^xB]$, for $y = 2$, and to $[L^xB^y]$, for $y \geq 3$. If the multiplicity factor is equal to or greater than 3, F operation produces a redundant output to Y operation. Certainly, both composer and analyst are free to choose which operation (Y or F) is more appropriate to connect the classes according to any of their purposes.

Toward a more refined organization of the tc-space, providing a general view of the operations that connects all textural classes, we propose the *textural class lattice* (TCL - Figure 3). The nodes of the graph are the textural classes, and the edges are operations that can connect the nodes. Each edge represents a different operation and the signs (+ or –) are reflected in their orientations. Double red lines indicate a relation between a pair of textural classes under the operation Y_B. Reading upward from left to right indicates the positive operation, and the opposite direction denotes the negative operation. Dotted green lines stand for Y_L operation, with upward from right to left designating $+Y_L$, and downward from left to right the $-Y_L$. Blue lines denote operation $+F$ from left to right (also considering the upward movements), and $-F$ from right to left (including downward movements).

Having the monopart classes, the simplest among the types, in the bottom of TCL reveals that all other classes derive from them through Y and F. Each monopart is connected to its correspondent polypart class by Y operation, which delimitates a spatial distribution of TCL based on the amount of Ls and Bs. Bottom-up and top-down movements, respectively, increase or decrease the number of textural parts (and, consequently, the number of threads within the class), while horizontal movements change the quality of the parts from lines (more to the left) to blocks (more to the right). Isopart classes balance lines and blocks in the middle of TCL. Except by isoparts, F operation connects both classes within all classes, but polyparts require a succession of F, with the isopart $[LB]$ as an intermediate point.

Let us return to Mozart's example (Figure 2) to examine the operations involved between the classes. Given that all classes have the single block $[B]$ in common, we can relate them to the first block in the violins through Y operation. First, the imitative process itself, under a textural perspective, consists of a positive Y operation to include a duplicated textural part. Thus, the first

[12] Y_L correspond to Gentil-Nunes' partitional operator named *revariance*, while Y_B can be interpreted as the *concurrence* (see [10, 46-49]).

[13] The operation $-Y_n$ (K) is context-sensitive because K must have a textural class n in it. For example, $-Y_L([B^y])$ is not defined, since $[B^y]$ has no line to be eliminated. The same principle can be observed in $-Y_B([L])$.

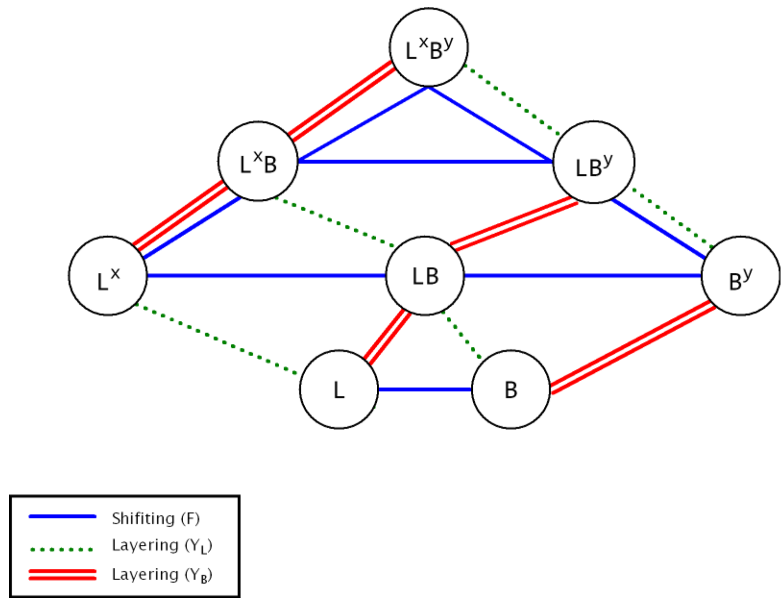


Figure 3: *textural class lattice: all textural classes connected by layering (Y) and shifting (F) operations.*

gesture corresponds to $+Y_B$ operation to connect classes $[B]$ to $[B^Y]$. The next linear transformation involves either splitting the block into two lines and merge one of the lines to the other block, increasing its thickness; however, in Tc-space, classes $[B^y]$ and $[LB]$ are related to $-F$ operation since the change in the thickness of the block does not affect the class representation. [14] A non-linear transformation reveals that class $[LB]$ derives from the initial block by $+Y_L$, a variation within the Y operation used in the first transformation. Indeed, the next two classes in the sequence successively articulate both Y_L operations (+ and -) to produces the oscillation between $[LB]$ and $[B]$ in the last measures.

ii. Unordered-Partition Space (up-space)

For a more detailed description of a texture we can ascribe an integer to express the actual amount of threads of each textural part[15]. While L stands for the integer 1, $[B]$ encompasses any number greater than or equal to 2, that is, according to the number of threads $[B]$ may refer to 2, 3, 4, 5 and forth on. In the same way, $[L^x]$ corresponds to x realizations of 1, whereas $[B^y]$ stands for a set of y integers greater than 1, which may include blocks with equivalent thickness or not. For example, for $y = 2$, that is, $[B^2]$ in a set of six threads only two combinations are possible: $[2, 4]$ and $[3, 3]$. This numerical representation is called *integer partitions*. Each number within a partition is also called textural part.

A partition can be defined as the various ways to represent a positive integer n through the sum of other positive integers [2, p. 1]. Each partition encodes a specific textural configuration in such a way that the texture of a given piece is represented as a sequence of partitions. Giving that the partition set for n is finite, using partitions allows the composer to have available all textures

[14] Another possible interpretation would consider a composite operation, such as $+Y_L$-Y_B or $-F$-Y_B, but, when it is possible, we prefer to use a single operation to simplify the process.

[15] Such methodology was first proposed by Wallace Berry [3] and further developed by Pauxy Gentil-Nunes [13][10] [9] through the approximation with the Theory of Integer Partitions, an important area of additive number theory developed by many important mathematicians, such as Euler, Hardy, Ramanujan, among others.

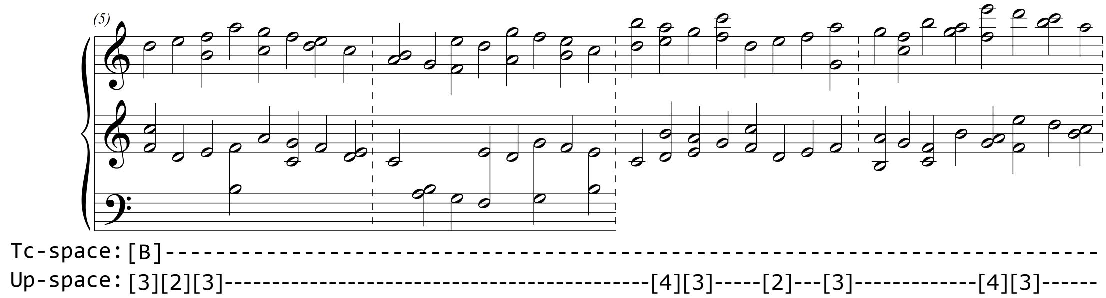


Figure 4: *Textural class [B] and Partitions encoding the thickness variations in the blocks of Ligeti's Étude 15: White on White, mm. 5-8 (1995).*

for the n number of threads, a significant tool for the compositional process.[16] As an example, four threads has five possible partitions: [4], $[1,3]$, $[2,2]$, $[1,1,2]$, and $[1,1,1,1]$.[17] These partitions are members, respectively, of the following textural classes: $[B]$, $[LB]$, $[B^y]$, $[L^xB]$, and $[L^x]$.

Within a partition the order of parts is irrelevant and, by convention in this work, the parts are presented in an increasing order. For a more concise presentation, the partitions are notated with square brackets in an abbreviated version, where the multiplicity is expressed by an index in a similar way than in textural classes. Moreover, in order to eliminate eventual notational ambiguities and excessive use of spaces, each part is separated by a comma or by the indexes of the previous part. For example, partitions $[1,1,1,2]$ and $[1,2,2,3,3]$ are, respectively, written as $[1^32]$ and $[1,2^23^3]$.

The partitions form a partially ordered set that constitute the second textural space, called *unordered-partition space* (*up-space*).[18] Compared to tc-space, up-space provides a more refined description of texture since the actual thickness and multiplicity of each part is depicted by an integer number. Moreover, partitions specify the textural morphology allowing the measurement of nuances among textures within the same textural class through the difference of their subtle internal variations.

For instance, in Ligeti's *Étude 15: White on White* (Figure 4) the chords produce a static texture made up of a sequence of various blocks. Tc-space simplifies the description of the texture ascribing the monopart class $[B]$. Thus, this abstraction does not regard the number of pitches within each chord as relevant. In up-space the texture is more detailed, being encoded through a varied progression that encompasses three partitions: [2], [3], and [4].[19]

Operations on up-space

Similar to tc-space, the operations that connect partitions in up-space concern changes in the multiplicity and thickness of a partition, but in a more specific way. Instead of the abstraction L and B, we use a positive integer to depict the exact number of threads we are including or

[16] This possibility is one of the most important contribution of Gentil-Nunes' Partitional Analysis to music composition.

[17] The sum of the threads of a partition corresponds to what Berry calls *density-number* [3, p. 188].

[18] By convention in this paper, the term partition refers to unordered partition. When the order is relevant it includes the term "ordered".

[19] We are considering the same onset of each thread as a criterion to determine the textural parts. Nevertheless, other relations can be considered in the analysis. For example, one could argue that the registral placement may emphasize some lines within the blocks, provoking a contraposition relation between them and the rest of the block. Or yet, that each staff should be understood as an individual textural part, and so on. In both cases, the analysis would include other textural classes and partitions.

excluding in the partition. In general, we can propose any relational operation to connect two or more partitions based on a given derivative process. To facilitate the discussion, here, we will focus on two basic operations that when combined are capable to connect all partition set: *layering* (Y) and *dimensioning* (D).[20] In contrast to tc-space, where the recursive application of the operations is limited, in up-space, both operations, when contextually applicable, can produce a path with unlimited chain of partitions provided by their successive iterations.

As in tc-space, layering operation refers to multiplicity either by including or excluding a textural part. The notation $*Y_n$, where "*" stands to either a + or – symbol, indicates the positive integer n that shall be included or excluded in the partition. Thus, the negative layering ($-Y_n$) is also contextual, demanding that the partition has a part equal to the integer n therein; otherwise, the operation would be not defined. Suppose $\{a, b, c\}$ are the parts of partition K. Thus, $+Y_d(K)$ is equal to $[a, b, c, d]$, while $-Y_b(K)$ correspond to $[a, c]$. Henceforth, we shall use Y to refer to $*Y_n$. Y is the inverse function of itself by inverting its signal. To put this another way, if Y is positive, to restore the input, we shall operate a negative Y in the output and vice versa. For example, $+Y_3([1^2 4])$ is equal to $[1^2 3, 4]$, and $-Y_3([1^2 3, 4])$ return the first input ($[1^2 4]$).

Dimensioning, a substitute of shifting in tc-space, indicates the increment or decrement of the number of threads of a textural part being noted as $*D_n$, where "*" is either a + or - symbol and n is the number of threads to be included or excluded from a part. Thus, $+D_2(K)$ (we say a positive dimensioning) consist of adding two threads in any part of K, increasing its thickness. Similarly, $-D_3(K)$ (a negative dimensioning) concerns the removal of three threads of any part of K, decreasing its thickness.[21] For example, $+D_1([1, 2, 3])$ stands for three possible outputs: $[2^2 3]$, $[1, 3^2]$, and $[1, 2, 4]$, while $-D_2([1, 2, 3])$ has a single output ($[1^2 2]$). Henceforth, we shall use D to refer to $*D_n$. As Y operation, D is also the inverse function of itself in an opposite direction, but, depending on the number of parts, its reverse operation may also produce outputs other than the input.

Note that the cardinality of n in Y and D refers to the distance between the input and the output partitions. This distance is related to the relative complexity of them [21]. The higher the cardinality of n, the more distant they are to each other. In the same way, if n is equal to 1, both operations reveal an adjacent relation between the involved partitions.[22]

When signed positively, both operations derive from the inclusion relation, in which the input partition is contained in the output. Also, while the recursive application D in any partition with two or more parts will necessarily produce bifurcation within the path, Y gives rise to a single and linear path. Using *Young's diagrams* makes these relations clearer.[23] Figure 5 demonstrates

[20] Despite their conceptual differences, both operations are related in some level to, respectively, Gentil-Nunes' *revariance* and *resizing* [10, pp. 45-50].

[21] Obviously, in the negative dimensioning, the altered part shall not be less than or equal to n because it would imply in its deletion, an effect that concerns the multiplicity instead of thickness.

[22] One of the main conceptual difference to Gentil-Nunes' proposal is that his partitional operators invariably considers one thread at a time for both transformations (resizing and revariance). Although the result of D for n greater than 1 can be equal to n successive applications of resizing, it does not cover all resizings' possible outputs because the n threads of D are necessarily included or excluded in the same part while resizing allows a balanced distribution of them among the parts. In Y operation, if n is greater than 1 it correspond to the composite operation of revariance and resizing instead of the n operations of revariance. However, although Y can be understood as operating on both textural dimensions, we consider the output as the insertion of a part of any thickness, a change in the multiplicity of the partition. Furthermore, while Gentil-Nunes' aim was to provide a topological study of partitions through their adjacencies to guarantee a consistent theoretical foundation, we are assuming a creative perspective for the operations, so that adding or excluding multiple threads simultaneously in both operations constitute a simple compositional procedure to alter the partition, regardless whether the output is adjacent to the input or not.

[23] Young's diagrams are visual ways of representing a partition. Each square corresponds to a different thread. Then, the partition is defined by the square's arrangements so that side-by-side squares stand for the thickness of a part and overlappings are related to their multiplicity [2, pp. 6-7].

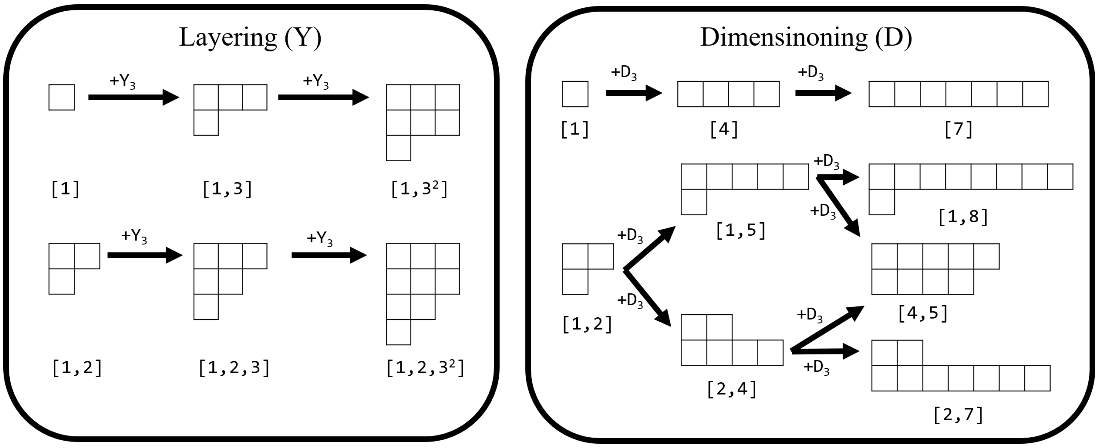


Figure 5: *Young's diagram showing the paths produced by Y (for $n = 3$) and D (for $n = 3$) applied to partitions* [1] *and* [1, 2]*.*

through Young's diagram the recursive application of $+D$ and $+Y$ for $n = 3$ departing both from a single-part partition ([1]) and from a multiple-part partition ([1, 2]). In multiple parts, the path of D includes bifurcations. Each partition can be included within the following one, which illustrates the inclusion relation commented above.

From an exhaustive taxonomy of the textural configurations, as well as a topology of the partitions operators, Gentil-Nunes [10, pp. 50-51][8, pp. 97-98] proposes the *Partitional Young Lattice* (PYL), based on Young's Lattice (Figure 6).[24] Each square represents a partition (or a pair of partitions as in density-number equals to 6) in such a way that partitions placed side-by-side in horizontal share density-number values. Moreover, similar to TCL, the more to the right are positioned the partitions, the more massive they are (more blocks). Conversely, the more to the left, more polyphonic are the partitions (more lines). Above each partition is stated its equivalent textural class for purpose of comparison. The color of Ls (blue) and Bs (red) expresses the division mentioned above. Any upward movement is expansive, increasing the number of threads of the partition. The number in parenthesis below each partition stands for the pair of indices called *agglomeration* and *dispersion*, whose functions are presenting the number of, respectively, collaboration and contraposition relation between all of threads within the partition, analyzed in pairs (see [10, pp. 33-38]). Each index increases or decreases according to the horizontal position of the partition since they are related to the polyphonic and massiveness degree. The edges that connect the partitions correspond to both operations (Y and D) for n equal to 1 to show the adjacencies of them. Partitions that do not share edges have no definite co-relations, due to the partially ordered set property of partitions.

In the partitions of number 6, the two pairs of partitions within the same square share the same pair of indices (agglomeration and dispersion). Thus, both partition can be understood as equivalent, due to their internal organization of threads, in which they hold the same relations of collaboration and contraposition despite the difference of their component parts.[25] This property, called the *h-relation*[26], becomes more recurrent, including more partitions within the indices, as

[24] Young's Lattice consist of a representation of the partition's lexical set using Young's diagrams, organized by the inclusion order, that is, bottom partitions are included in upper partitions.

[25] The musical impact of this equivalent is left for a forthcoming work.

[26] The "h", which stands for homoindex, is an allusion to Allen Forte z-relation, where two pitch-class sets not related by transposition or inversion produces the same interval-class vector [6].

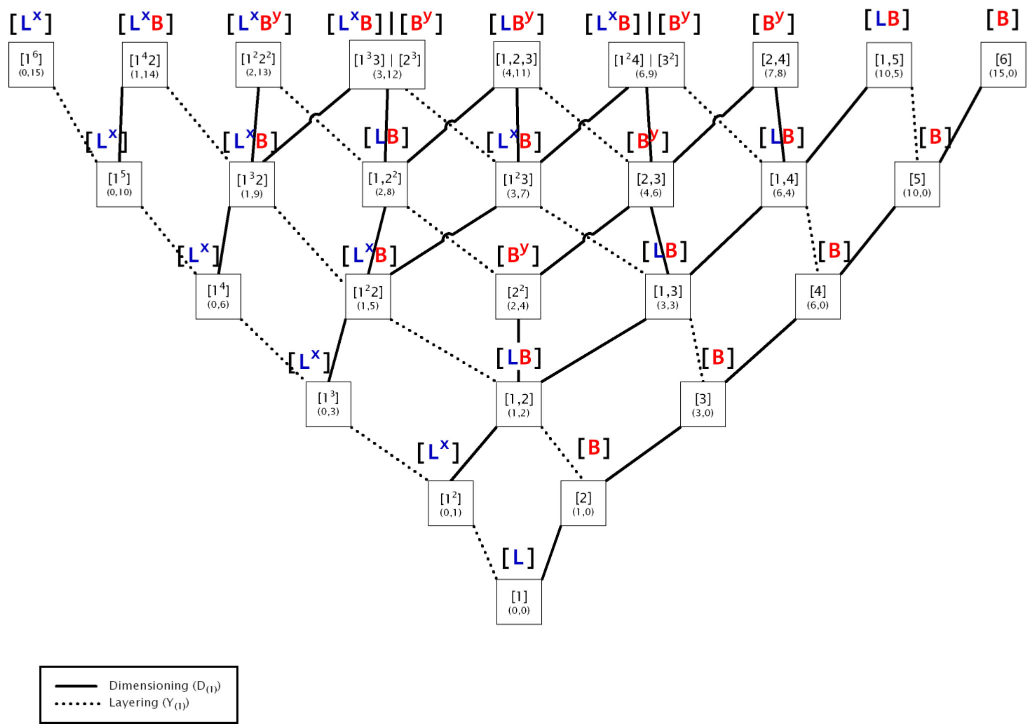


Figure 6: *Partitional Young Lattice: partition lexical set for density-number = 6 connected by layering (Y_n - straight lines) and dimensioning (D_n - dotted lines) for n = 1. (adapted from [10, p. 51]).*

the density-number increases. If we examine the partitions of the number 15, for example, within the 176 partitions there are 42 pairs of indices that include h relations from two to six partitions h per index. Thus, most of the partitions of number 15 are h-related to another [21].

Now, let us examine how the operations allow us to understand the way partitions evolve from one to another. In Debussy's *Prelude VIII*, *La Fille aux Cheveux de Lin*, the texture can be encoded by partition sequence: <[1][1,4][1][5][6][5][2^2][1,2][1,4][6][7][1]> (Figure 7).[27] The excerpt consists of two arc-shaped gestures around the line (partition [1]). The first arc (mm. 1-4) is a simple deviation of the line through the inclusion and exclusion of the block [4] (a $+Y_4([1])$ followed by $-Y_4([1,4])$). For the second arc (mm. 4.3-7) Y and D are combined to build other curves within the arc in a more elaborate gesture.

First, the line becomes the block [5] by including four threads to increase its thickness (noted as $+D_4([1])$ = [5]). Given the number of threads involved, this transformation can be understood as a variation of the Y in the first arc but assuming a collaborative relation with the line instead of contraposition. After that, D produces a small arc or oscillation in the block [5] through insertion and exclusion of a single thread thereinto ($+D_1(5)$ and $-D_1(6)$). Furthermore, the transition from partition [5] to [2^2] demands, necessarily, the combination of Y and D with opposite signals, so that the block's thickness is decreased by three threads ($-D_3$), and the part {2} is added to the partition ($+Y_2$). The following partition $[1,2]$ outcomes from the exclusion of one thread ($-D_1([2^2])$). Finally, except by partitions $[1,4]$ and [6] that involves both operations Y and D, the closure of the arc concerns only thickness transformation of the parts. Note that the maximum of threads within a partition (7) occurs in the last part of the second arc, which is suddenly decreased to form the

[27] The criterion to ascribe each partition considers the evaluation of rhythmic coincidence.

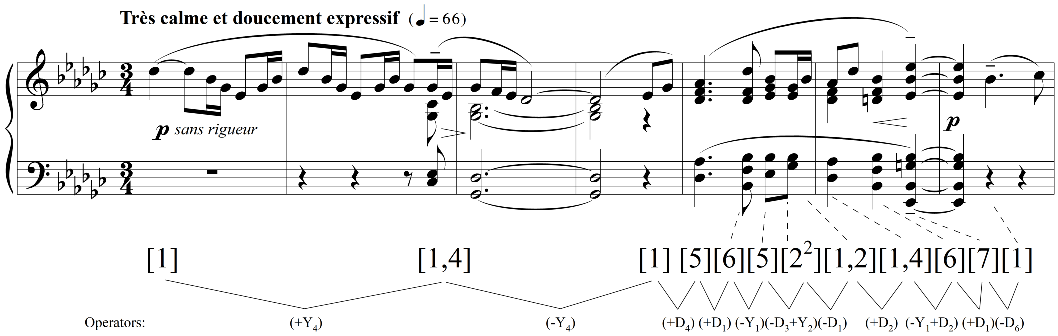


Figure 7: *Partitions that encode the texture sequence of Debussy's Prelude VIII, La Fille aux Cheveux de Lin, mm. 1-7 (1910), and the operators that connect them.*

line [1], resulting in the largest gap between any consecutive partitions. Given that the first pitch in partition [1] is also within the chord in partition [7], this transformation could be understood as a "filtering process", in which all but one pitch is preserved to produce the next partition. Perhaps, this procedure is a strategy to smooth the textural gap in the chain by approximating the musical content of them.

Another way to think about the texture of this piece is to consider each piano hand as autonomous. Thus, the texture depends on the way the threads are divided into each staff of the score according to the rhythm coincidence, so that each hand produces a individual part and the partition is the union of them. The combined sequence (<$[1][1,2^2][1][2,3][3^2][2,3][2^2][1,2][1,2^2][3^2][1]$>) reveals that the left hand is more stable for alternating among blocks with two, three, and four threads. In contrast, the right hand combines three parts ([1], [2], and [3]) to produce a more elaborated sequence. This combination does not exceed three threats at a time.

Of course, the description through partitions do not cover every aspect of texture; many attributes thereof cannot be encoded into a partition. This is especially clear considering that within a partition the order of the parts is irrelevant, then any characteristic of a texture that involves a vertical spacialization, for example, is out the context of partitions. Figure 8 presents an example of this limitation. The partition $[2^2]$ is defined by the timbre in such a way that either *pizzicato* (blue squares) or *arco* (red squares) assembles one of the blocks 2. Also, the pitch content helps to demarcate the blocks. All pizzicato parts are formed by members of trichord 3-5[016] while the arco parts comprise members of trichord 3-3[014][28]. Although partition $[2^2]$ expresses the general organization of the parts, it does not describe the multiple combinations among the instruments as an elaboration within the partition. Thus, up-space omits some textural data, which can undermine an analytical inference. Surely, under the creative perspective, the lack of details allows various compositional fruition, but, at same time, by refining the texture, the design of a composition can involve more advanced connections. In order to develop this approach, we shall include a vertical ordering to the parts within a partition as a way to depict either their registral placement or timbre distribution.[29] Thus, the last textural space concerns the ordering within partitions.

[28] See [6].

[29] Berry also discusses the relations between texture and register, but his focus is on the actual vertical span of textures, that is, the interval among their uttermost part, and the way such space is fulfilled by the threads (called as *density-compression* [3].

Figure 8: *Different organizations of threads within partition [2^2] according to combinations of* pizzicato *(blue squares) and* arco *(red squares) effect.*

iii. Partition Layout Space (pl-space)

The spatial factor of texture is an important aspect to differentiate the realization of the same partition within two or more different contexts without necessarily dealing with their actual musical content.[30] We can order the parts within a musical partition by either register placement or timbre. If we consider the register, each textural part is ordered in the present paper according to their spans as a result of the unfolding of the music materials into a sounding medium. Else if we consider timbre, then the order can be read top-down from the score disposition. Once we identify whether a part or thread is higher, equal or lower than another through register or timbre, we can ascribe a *partition layout*.[31]

A partition layout consists of a more precise way of encoding a texture by expressing details regarding the internal organization of the partition's component parts. To depict this order we propose two types of components within partition layouts related to a specific notation: an integer number in the form of a *ordered partitions* and threads or grouping of threads, noted as a *thread-word*.

An *ordered partition* can be defined as a partition wherein the order of parts matters[2, p. 54].[32] Hence, a given partition can provide different ordered partitions according to the possible permutation of its parts. For each positive integer n there are 2^{n-1} possible ordered partitions. For example, the number 4, in addition to its five partitions, presents three ordered partitions, resulting in eight configurations: <4>, <1,3>, <3,1>, <2^2>, <$1^2 2$>, <1,2,1>, <2,1^2>, and <1^4>.

[30] By spatial factor we are not referring to what Berry calls *texture-space*. While Berry's concept involves the measurement of the actual registral boundaries of texture, we are concerned to the relative vertical distribution of the parts.

[31] This comparison among the parts to define their relative position is similar to the process involved within the contour spaces ([25]).

[32] In mathematics, an ordered partition is called *composition*, but in the present work that word is avoided for the sake of clarity (composition is already used as a musical concept).

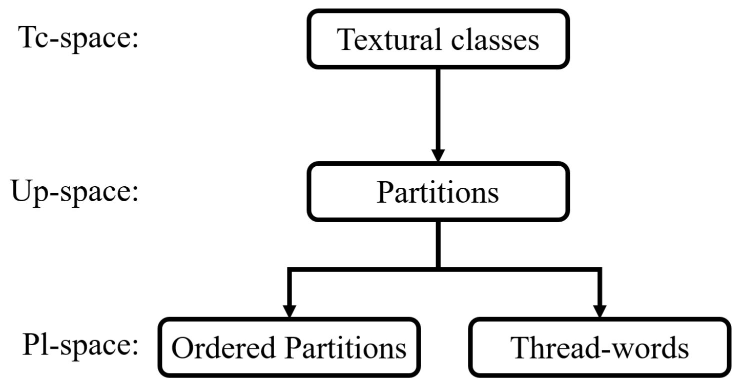


Figure 9: *Inclusion relations among the classes within textural spaces.*

Note that the applicability of ordered partition demands a stratified textural organization, that is, the overlapping of layers (or strata) shall be well-defined; the forming threads of a block shall not be vertically interpolated with any other part. However, most often, the parts may cross the vertical boundaries, connecting apart threads into the same textural part. An ordered partition cannot express a texture whose threads are "interwoven" in the register.

The actual registral placement of non-contiguous organizations (interpolated parts) are described by an organization called *thread-word*. A thread-word is a representation in which each part of a texture is identified by a letter in such a way that each individual thread receives a different letter, and threads that assemble a block are ascribed with the same letter. Thus, the number of letters indicates the cardinality of texture (density-number), the number of occurrences of a given letter is associated with the thickness of that part, and the number of parts is deduced from the variety of letters. For example, a thread-word $<a, b, c, d>$ stands for partition $[1^4]$, given that there are three different letters (threads) without duplication while a thread-word $<a^2b^5>$ indicates two blocks {2} (noted as $\{a^2\}$) and {5} (noted as $\{b^5\}$). The use of a thread-word allows us to observe interpolated part, as in thread-word $<a, b^2, a^2>$. In this case, the five threads are grouped in an interwoven texture with two distinct parts (indicated by letters a and b). This texture is a layout of partition $[2, 3]$ in which part {2} (indicated by b^2) is interpolated with part {3} (the sum of components $\{a\}$ and $\{a^2\}$).

Both proposals (ordered partitions and thread-words), as sub-classes of partitions, provide further information for up-space. Similarly, partitions can be understood as sub-classes of textural classes. Also, textural classes are class containers that encompass partitions in up-space. For example, textural class $[LB]$ comprises, among others, partition $[1, 2]$, and all its possible partition layouts (ordered partitions ($<1, 2>$, $<2, 1>$) and thread-words ($<a, b^2>$, $<a, b, a>$ and $<a^2b>$). Figure 9 shows this relation of textural spaces from the most superficial (tc-space) to the most specific (tl-space).

Although we could use Berry's methodology, in which the textures are written vertically, with their parts stacked according to their vertical position [3], either ordered partitions and thread-words are written horizontally within "< >", following the ordering from the highest to the lowest registral placement or timbre disposition of the parts, to clarify and facilitate the visualization (and notation) of the parts' ordering. Thus, in a thread-word, the first letter is always "a", that correspond to the upper thread in texture. For each new part is ascribed another letter following the alphabet. When there are two equal contiguous letters, we use the abbreviated

Figure 10: *Layering (Ordered Partitions encoding the texture of Bartok' string quartet No. 2, Op. 17, mm. 1-4 (1920).*

form with an exponent.[33] For example, partition [1, 4] provides two ordered partitions <1, 4> and <4, 1> to indicate the exact position of the line (part {1}) and block {4}. In addition to these two layouts, thread-words include other three organizations: $<a, b, a^3>$, $<a^2b, a^2>$, and $<a^3b, a>$. The threads ascribed by "a" assemble the block {4}, while "b" refers to the line.[34] Surely, the register evaluation shall not be totally strict since any eventual crossing parts would be computed as a different ordered partition. The set of all partition layouts (ordered partitions and thread-word) forms the *partition layout space* (*pl-space*), the most precise encoding of a texture among all textural spaces.[35]

Figure 10 shows an example of ordered partition providing an analysis of an excerpt of Bartok's *String Quartet No. 2*, Op. 17 (1920). The sequence starts with the cello solo presenting partition [1], followed by the block [2] assembled by second violin and viola, and, finally, another line in the first violin [1]. In this case, both ordering criteria (register and timbre) provide an equivalent result. The partition chain $<[1][1,2][1^2 2]>$, even though encoding the threads organization of texture, it ignores the spatial organization of threads, an important detail of the texture. Thus, by examining the ordered partition, we notice that the lines are in the uttermost voices, enclosing the block. Also, the chain <1><2, 1><1, 2, 1> indicates that the texture is made up of the accumulation of new parts. That is, each partition is contained in the following one, preserving their registral (or timbric) disposition. This process expands upward the vertical space of texture, a property that is not encoded by the representation of unordered partitions.

To see how the information provided by thread-words can be essential for the understanding of textural functionality, let us examine an excerpt of Varèse's *Octandre* (Figure 11). The texture can be encoded by three partitions ([3, 4], [7], and [1, 7]) that set up two pitch and dynamic contents according to the number of threads therein. Within eight threads, the clarinet holds a line that is apart from the block of the other instruments through rhythmic and dynamic opposition. The seven-thread textures derive precisely from the pause of the clarinet, as we can observe in the

[33] Attention that in a thread-word the exponent refers to the thickness of the component part instead of its multiplicity. Thus, $<a^3>$, for example, expresses a block with three contiguous threads instead of three lines, which would be noted as $<a, b, c>$.

[34] We are not considering the redundancies between ordered partitions and thread-word, as, for example, $<a^4, b>$ or $<a, b^4>$. See section III for a discussion about the possible realizations of a partition.

[35] A step farther than pl-space would result in either a redundancy of the music score itself or the description of the particularities within each part (their musical content).

Figure 11: *Layering (Varèse's Octandre (1923), mm. 50-54: unordered partitions and thread-words encoding the texture.*

score. By examining partitions $[1,7]$ and $[7]$, these relations also become clearer. Partition $[3,4]$ indicates a split variation of the block [7]. If we use the thread-words, more details regarding the texture will be explicit.

While partition [7] does not enable multiples realizations, partition $[1,7]$ is depicted as $<a^2b, a^5>$ according to the score order. This representation reveals that the clarinet is within the block, dividing the seven threads into two groups (2 and 5, as indicated by the exponents). Without accessing the score, the thread-words indicate the relation between the cardinalities through the clarinet, given that $<a^7>$ and $<a^2b, a^5>$ have the same number of threads assembling the part $\{a\}$ without the interpolated part $\{b\}$, a possible exclusion of it from the texture. Furthermore, while partition $[3,4]$ does not present any relation with other but the number of threads (in this case, 7), $<a^3b, a, b^2>$ shows the amalgam of threads in both blocks, suggesting a mix between the interpolated $[1,7]$ and the block [7]. In fact, by comparing both structures, this relation becomes still clearer. Both organizations have an isolated thread in the center being surrounded by a group of combined threads. That is, while part $\{b\}$ divides $\{a^2\}$ and $\{a^5\}$ in $<a^2b, a^5>$, the combined parts $\{a\}$ and $\{b\}$ are surrounded by $\{a^3\}$ and $\{b^2\}$ in $<a^3b, a, b^2>$. This shows that thread-words can give access to critical information that in some cases can be more useful than the simple enumeration of partitions.

Operations on pl-space

Operating on pl-space demands the observation not only about thickness and multiplicity of the parts, but also about their order. Thus, in addition to operations Y and D, inherited from up-space, pl-space includes two new operations to modify the order, called *mirroring* (M) and *permutation*

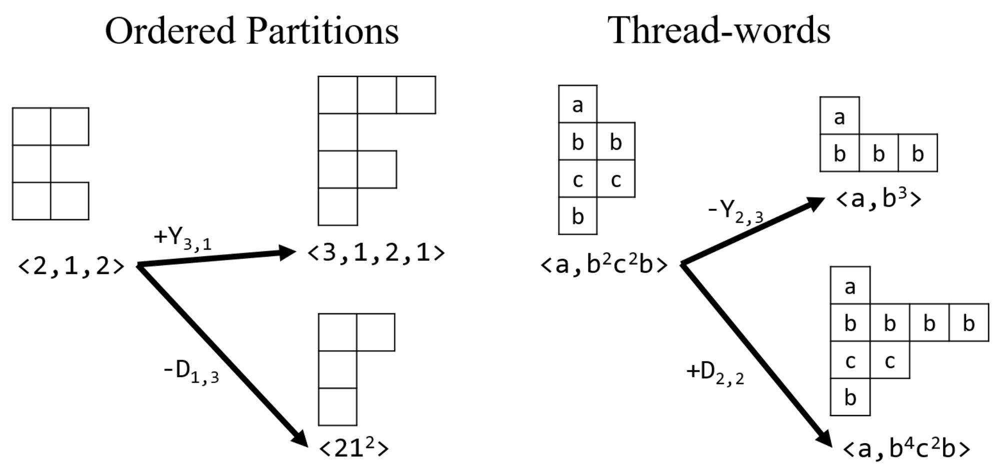


Figure 12: *Ordered Young's diagrams demonstrating Y and D operations including order in ordered partitions and thread-words.*

(T). Both operations do not have a + or - signal because no thread is added or removed from the partition layout.

Concerning order, each position within an ordered partition is numbered from 1 up to the number of parts or threads in the partition layout. Let us say that $\{x, y, z\}$ are the components of a partition layout K. The parts x, y, and z are either an integer number or a letter that refers to a thread. The part $\{x\}$, $\{y\}$, and $\{z\}$ are, respectively, in positions 1, 2, and 3. Thus, any operation involving the order of parts of K will include a reference to position 1, 2, or 3. In thread-words, the position refers to the letters regardless to their exponents in such a way that the third position of $<a^2, b^4, c^3>$ is $\{c^3\}$. Obviously, these operations are only effective in partition layouts with two or more components.

Both operations from up-space (layering and dimensioning) can be applied to pl-space to alter a specific component by appointing its position. The notation is $*X_{n,m}$, where X is either Y or D operation, "*" stands for either + or - signal, n indicates the number of threads to be included or excluded from the partition layout, and m refers to the exact position of this part. Thus, $+D_{n,2}$ applied to a partition K (noted, $+D_{n,2}(K)$) is equal to the inclusion of n threads in the part located in position 2, increasing its thickness, while $-D_{n,4}(K)$ consist of the exclusion of n threads of the part in position 4. Similarly, $+Y_{n,3}$ includes a part with n threads in position 3, and $-Y_{n,5}$ excludes the component in position 5. Note that in both partition layouts, Y operation requires that n shall be equal to the component to be excluded. Also, in thread-words, -Y shall not operate in parts that is a member of a thicker part because it would affect the thickness instead of multiplicity. Similarly, $*Y_{n,1}(K)$ will either increase or decrease the thickness of the first component by n threads. It is important to remember that in -Y, n shall not be greater than or equal to the part to be altered because it would affect the multiplicity. For instance, if we operate $+Y_{3,1}$ and $-D_{1,3}$ on ordered partition $<2, 1, 2>$ the outcomes are, respectively, $<3, 2, 1, 2>$ and $<2, 1^2>$. Note that in Y the part $\{3\}$ is inserted in the first position. Thus, the first position of the input becomes the second position of the outcome. In the same way, $-Y_{2,3}(< a, b^2c^2b >)$ is equal to $<a, b^3>$ and $+D_{2,2}(< a, b^2c^4b >)$ stands for $<a, b^4c^2b>$. For a better visualization of these operations, we can use an ordered version of Young's diagram in which the vertical position reflects the order of parts. In order to depict the thread-words, we include a letter inside the square to indicate the corresponding threads.

Mirroring is the operation that flips the order of the components parts in such a way that the bottom (or the last, considering the horizontal notation) becomes the upper (or first) and vice versa. This operation is a one-to-one mapping of components around an inversional axis, that is, each component of a side of the axis is swapped with its corresponding position in the other side. If there is an odd number of component parts, the axis is the median component. If there is an even number of components, then the axis point is located between the two median components, causing their swapping along with the other ones. For example, given two ordered partitions $K = \{3,1,2,1,5\}$ and $J = \{2,3,1,3\}$. The mirroring of K (noted as $M(K)$) consist of swapping all parts around part {2} so that $I(K)$ is equal to <5, 1, 2, 1, 3>. Similarly, $M(J)$ swaps the parts around the axis formed by parts {3} and {1}, producing the ordered partition <3, 1, 3, 2>.[36] In the same way, in thread-words, $M(< a, b^2a, c^2 >)$ is equal to $<a^2b, c^2b>$ (a rewritten version of $<c^2a, b^2a>$).

Permutation noted as $T_{n,m}$, where n and m indicate two different positions in the partition layout, consisting of interchange (or swap) the positions of two components. Let us say that the partition layout K encompasses the parts $\{x, y, z, w\}$, where x, y, z, and w are either an integer or a letter ascribing a thread. Then, $T_{1,4}$ means that the part in position 1 and 4 shall permute their position to each other in such a way that $T_{1,4}(K) = <w, y, z, x>$. For example, $T_{2,3}$ <1, 2, 5, 2> is equal to $<1, 5, 2^2>$. Similarly, the outcome of $T_{1,4}$ $<a^2, b, a, b^3>$ swaps the components $\{a^2\}$ and $\{b^3\}$, producing $<b^4a^3>$; however, when $\{b^3\}$ assumes the first position, it shall become an "a". Thus, the outcome is rewritten as $<a^4, b^3>$. If the partition layout has only two parts, then permutation is equal to inversion.

III. Compositional Perspectives of Textural Spaces

Up to now, we have concentrated on the definition of textural spaces, and the basic operations associated with each one of them. This definition is important to establish the theoretical ground necessary to understand the multiple features of texture. Now, we will discuss some aspects of how these spaces can be articulated within a composition, highlighting part of their compositional potential. Also, we will discuss the way a texture can be decoded into a music score.

In general, the combination of parts (lines and blocks) in any space suggests a hierarchy of the type *figure/ground* [26]. Blocks tend to be in the background, supporting the lines in the foreground. However, this logic was often inverted, principally during the twentieth century, as a replacement to the traditional melodic composition. Blocks that were usually treated as accompanying textures became the main prominent musical idea.

In tc-space, due to the linear presentation of solely one part, monopart textures has a sense of a narrative analogous to a single speech. This lack of interactions with other parts assigns to the texture a secondary role, giving focus to other musical parameters. In fact, monopart classes may be (and often are) used as a compositional strategy to highlight a melodic idea, a particular rhythm, a harmonic sonority, etc.

On the other hand, overlapped parts in other types include a contraposition relation among the threads, which may affect the overall complexity. Thus, when a texture has concurrent parts it shifts from a narrative to a creative structure, assuming an important role to unfold the music form, a ploy often used by twentieth-century composers as an alternative to overcome pitch dominance. Thus, a textural sequence built up only from monopart textures may limit the creative use of texture. Its importance, then, is decreased in the compositional process, whereas using other types of tc-space in combinations (including monoparts) or isolated may offer various compositional

[36] Although there are similarity between M operation and the retrogradation of pitch domain, we are considering the vertical alignment of the parts instead of their linear order. Moreover, a retrogradation operation can be applied to a chain of partitions or partition layouts to reverse the sequence of adjacencies, an operation that is out of the scope of this work.

solutions to unfold the music form. The same principle can be applied to the other spaces. In fact, by using up- or tl-space, the textural progression can be more creative and imaginative, allowing the design of a more sophisticated textural structure that includes multiple relations between and among their components. Thus, the importance of texture within the compositional process may be increased.

i. Defining a texture

The composer might ask "how to choose which textural space or which textures to use in a composition?" or "how to define which criteria to use to select textures?". To answer those questions we can consider connections between textures through operations, parsimony between them, the potential of musical implementation of these textures, among others.

Both TCL and PYL constitute a *compositional space* in textural domain.[37] This means that they can be observed as a game board ([8]) whereby the composer can produce various patterns of movements within a given number of threads by following the edges that connect textural classes and partitions to produce chains of textures. By defining a starting and an ending point, the composer may have a finite number of paths to connect them, considering, for example, the most minimal connection (the shortest path) or the path with more intermediate textures between them (the longest path). If two partitions within the path are connected by a single edge, then the movement between them is parsimonious, that is, the transformation from one to another can be smooth, a property that may be convenient within the same formal section. On the other hand, non-contiguous partitions that are connected by multiple operations may create contrast, which can be a divisory point in texture for a formal delimitation.

Although the operations we discussed so far allow us to transform a partition into another parsimoniously, these operations can be quite abstract since they do not explicit the potential process involved from each autonomous part or thread to evolve toward the next texture. Within tl-space, Y and D operations include the position of the part to be altered, which details the process involved. However, other potential transformations between successive textures can be acknowledge in this diachronic perspective. Thus, we can examine the role of each component in the unfolding of the next partition through the way they are connected (a "texture-leading"). The potential transformation according to the texture-leading of parts depends on whether the number of parts and threads will be altered or not. The preservation of the number of parts necessarily involves one of the following transformations:

1. An increment of decrement of the thickness of one or more components (D operation);
2. The reorganization of its internal components (M and/or I operations);
3. The combination of either M and/or I with D operation to change the order and thickness at once.

However, to increase or decrease the number of parts, in addition to the application of Y operation to insert or exclude one or more components, the transformations involve *merging* or *splitting* of two or more parts within the texture combined to a change in the thickness. Merging or splitting a component can be understood as the combination of Y and D operations in opposite signals (i.e., $+Y\text{-}S$ or $-Y+S$) in such a way that the number of threads is not altered (density-number).[38] When two or more parts are combined to produce a new part whose the cardinality

[37] Compositional Space is an out-of-time set of musical objects related by a given criterion. Such a structure provide the foundation for a compositional designs or an improvisation ([24, p. 336].

[38] This composite operations of Y and D is similar, at a certain level, to Gentil-Nunes *transference* ([10, pp. 47-48]).

is the sum of them, we say they were merged. When a part is divided into two or more parts, we say the part was splitted.[39] Both merging and splitting, when applied in isolation have an invariable density-number, that is, the number of threads in the texture remains the same. Other transformations may include the combination of two or more aforementioned operations. All of these linkage strategies are essential to composition (and analysis). In fact, although these procedures necessarily affect the texture, they are understood as a response to the manipulation of the other parameters (mainly pitch and rhythm).

Another way of creating a parsimonious texture-leading consists of examining the possible intersections between two consecutive textures, that is, one or more common part that they share that can be preserved from a texture to another. This intersection can be minimum, when all parts but one are different in both partitions (their intersection is equal to one), or maximum, if all parts of the first texture are contained within the next one. Two consecutive textures that have intersections of their component parts allow the composer to coordinate their connection in a smooth way.[40] A common use of intersection in musical repertoires involves Y operation, in which a complex texture is constructed by cumulative superposition of simple (or less complex) parts all contained therein.[41]

In pl-space the intersection of two consecutive partition layouts, in addition to the simple common-part procedure, may regard the ordered intersection. That is, if at least one part or thread of any partition layout is equal to the other, they can be arranged in such a way that the common element between them can be maintained in both textures in the same vertical position (registral span).

Using thread-words, intersections can be either literal, when the exact number of threads of a part is preserved into the subsequent texture, or abstract, if the threads are embedded in thicker parts of the other. In both cases, we shall not compare the letters but the number of their threads and their positions. For example, thread-words $<a, b^2a>$ and $<a, b^4c^2a>$ can be literally intersected by threads "a", considering that there are two of them in both textures, and by part $\{b^2\}$.[42] Moreover, part $\{b^2\}$ can also be embedded into part $\{b^4\}$ (an abstract intersection).

For a total intersection between two distinct partition layouts, we shall consider their alignment to examine whether it is possible to connect sequentially their components or not. Figure 13 shows an example of this embedding property by using ordered Young's diagrams. The shaded squares in $<1, 2, 3, 2, 1, 2, 1>$ and in $<a^2b, c^2, b, a^3>$ express, respectively, that $<2, 3, 1^2>$ and $<a, b, c^3>$ are totally contained into them. Attention must be paid to the fact that, although other literal and abstract intersections could be considered in other positions within a thread-word, only the case in question allows the total intersection since each part shall be ascribed by a different letter regardless the number of squares.[43]

The choice of a texture may consider its degree of *entropy*. According to *information entropy* ([32]), the complexity of a process can be measured by how much information is produced (or allowed to be produced) thereby. Thus, the higher the number of potential outputs of a process, the more complex it will be due to its high degree of unpredictability (entropy). In the same

[39] A precise formalization of merging and splitting, as well a further reflection of their effects are out of the scope of this work.

[40] Of course, only the intersectional property between textures does not guarantee the parsimonious sense since it also depends on the musical content of both textures (pitch, rhythm, dynamic, setup of timbre, registral placement, etc.).

[41] This staggered entrance of parts (or threads) is usually associated to the beginning of a piece or a section, as, for example, in the exposition of a fugue. In jazz music, such a procedure is called *pyramid presentation* (see [28, p. 17]).

[42] Note that in this case, we understand that $\{b^2\}$ is rewritten as $\{c^2\}$ because of the another part interpolated $\{b^4\}$.

[43] There are some ways to calculate if two ordered partitions have the embedding property between them. A further discussion about it would be more complex, and, therefore, it is out of the scope of this paper. For more information, see [14][5].

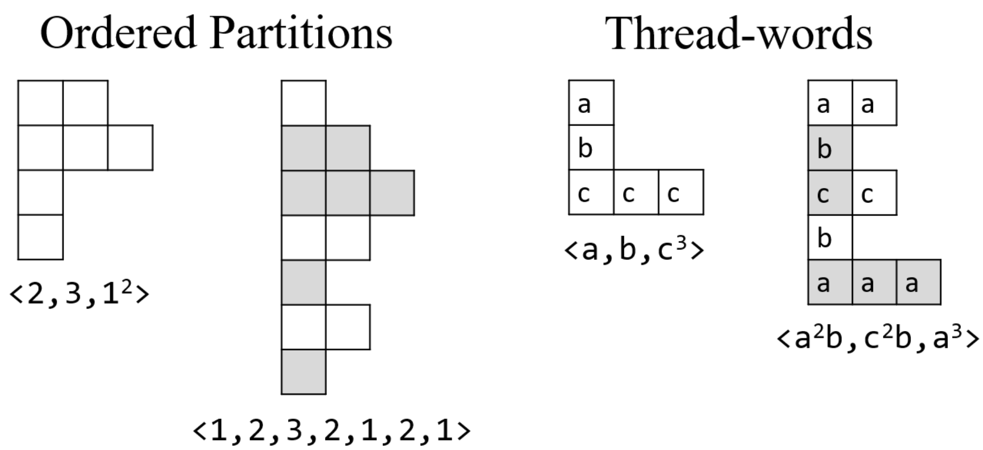


Figure 13: *Layering (Ordered Young's diagrams demonstrating the intersection property in ordered partitions and thread-words.*

way, a process that encompasses a small number of potential outputs are more predictable and, consequently, simpler. Within musical texture, we assume that the entropy level refers to the number of the potential implementations of a partition, that is, how many partition layouts a partition can produce. This is a significant aspect to be considered during the compositional work, after all, the more implementations a texture provides, more liberty the composer may have to explore creative solutions.

The entropy is therefore conditioned to the number of different partition layouts a texture can have. For example, partition $[1^3]$ has four ways to be implemented according to the internal organization of its threads: $<a, b^3>$, $<a^3b>$, $<a, b, a^2>$, and $<a^2b, a>$).[44] The thread-words that refer to the same partition are called in this paper as *anagrams*. Table 1 presents the number of anagrams for all partitions with one up to four threads. Note that any partition with a single part has only one possible anagram because within a block the contiguous threads cannot be differentiated. Similarly, given that the first component of any thread-word is always "a", the permutation between lines in the first position are also redundant since any line placed in the first position will be rewritten as "a". Also, the letters are included in an increasing order. Thus, any partition with multiple lines also has only one anagram as the form $<a, b, ..., z>$, where z stands for the number of lines therein, ascribed according to the alphabet order. Moreover, threads that assemble a block are understood as an internal repetition within a thread-word; therefore, the repetitions shall be eliminated from the number of possible permutations.

Furthermore, the entropy is also affected by relation between the density-number of the partition and the total number of available threads to decode it. That is, if the density number of a partition is lesser than the number of available threads for the composer, the entropy is higher, even in partitions with a small number of anagrams since the composer can decide which threads will be involved in the decoding process. For example, suppose that we have three threads available $\{a\}$, $\{b\}$, and $\{c\}$ to decode partition [2]. Although partition [2] has just one possible anagram, within three threads the composer have three possible ways to decode this into the musical score: $\{a + b\}$, $\{a + c\}$, and $\{b + c\}$.

[44]Note that $<a, b^3>$ and $<a^3b>$ correspond, respectively, to the ordered partitions $<1, 3>$ and $<3, 1>$.

Table 1: *Anagrams of partition lexical set for $n = 4$.*

Partition	Density-number	Number of Anagrams
$[1]$	1	1
$[2]$	2	1
$[1^2]$	2	1
$[3]$	3	1
$[1,2]$	3	3
$[1^3]$	3	1
$[4]$	4	1
$[1,3]$	4	4
$[2^2]$	4	3
$[1^2 2]$	4	6
$[1^4]$	4	1

ii. How to decode a texture into the score?

Once the composer decide which textures to use, the next step may involve its musical decoding (or realization). The musical realization of a texture can direct the composer's goals, allowing not only the approximation to different aesthetics and creative goals since the materials to be project into the textural structure is not defined. Mapping some of these possible realizations can help us to understand the musical implications related to a texture.[45]

In order to consider multiple possible decoding of a texture, we shall consider both the size of the window of observation and the possible partition used as a reference, called *referential partition*. Furthermore, the way in which a partition is decoded into the score is intrinsically related to the manipulation of musical parameters in the compositional process. Thus, departing from the standard ways of composing musical texture and form involved in composition, we initiate this discussion by defining a *standard mode* to decode a texture.[46]

By defining a standard mode of decoding enable us to understand the potential variations within a same texture. A standard realization is the most elementary and objective way to articulate a texture in music. One simply articulates the textural parts in some obvious musical way, such as a group of voices, a short uniform passage, or a block of sound. In analysis, standard realization is defined by the smallest possible *window of observation*[47] to encode a texture, with no global considerations. Thus, this mode implies a series of constraints.

First, all threads and parts of the partition shall be articulated simultaneously and continuously.[48] Either each part has the same onsets or some parts tie a note over a onset in another part. Any pause would result in a new partition, as a subset of it. Second, the chosen criterion for grouping the threads shall be preserved throughout the partition's duration. Otherwise, any change in one of the threads could produce either their contraposition or a collaboration with other threads, forming a new partition. Third, doubling threads by octaves, unison, or other

[45]Obviously, although we are proposing a formalization of the creative process within the textural spaces, the formation of a texture, in the current practice, can be combined with other aspects of musical realization, in a less systematically application.

[46]In order to facilitate the discussion, our discussion here will concern only the up-space, but all involved principles can be adapted to the other the textural spaces according to the composer's goal.

[47]The window of textural observation is an important feature of textural morphology that enables to assemble the threads in a temporal and abstract space, that can be as simple as a single simultaneity to large sections, as components of the same texture [11].

[48]Although we are considering the partitions to discuss the standard realization of textures, the discussion can be extended to textural classes and partition layouts since they are intrinsically related to each other.

Figure 14: *My Sagração de um Fauno na Primavera (2016/2019), mm. 5-9, coding the partition sequence <$[1^3][1,5][3^2][1^2 3][1,2,3][1,3][3][2][1]$> into the score.*

intervals alters the partition because it affect the thickness of the parts. In this case, a textural realization with larger ensembles shall consider doubling as part of the partition, which will result in partitions with a large number of threads and parts.

Standard realization is a good (and even didactic) way of composing from small textural units. For example, in the creative process of my composition, *Sagração de um Fauno na Primavera* for flute, bassoon and piano, the partition chain was articulated by standard realization considering rhythm alignment as the main criterion to define the collaboration and contraposition relations. Each pause or rhythmic coincidence was placed to articulate a new partition; therefore, the music is textually diverse, with multiple changes in a short temporal span. The squares encompassing the partitions below the score reveal the size of each window of observation (Figure 14).

To preserve the sense of good continuity between consecutive partitions, I applied some of the aforementioned strategies. First, when two consecutive partitions have intersections, I preserve the common part (or parts) in the same combinations of thread. For example, the bassoon line is sustained in the pair $[1^3]$ and $[1,5]$ (m. 5) and in the set <$[1^2 3][1,2,3][1^2 3]$> (mm. 6-7). Similarly, the piano's block [3] (mm. 6-7) is also preserved throughout the sequence <$[3^2][1^3][1,2,3][1^2 3][1,3]$>.[49]

Second, the potential of briefly merging parts to produce a block. As an example, see the bassoon line (mm. 5-7), in which the independence of the line is momentarily discontinued (m. 6) to form the partition $[3^2]$ through the rhythmic coincidence with piano right hand.

Third, consider different blocks of successive partitions as variations of the same threads through *D* operation. Thus, if a block has its thickness increased, we can simply add threads to it. Conversely, if the thickness is decreased, the threads can be removed from it. For instance, in the chain <[3][2][1]> (mm. 5-6), the decreasing process results from the suppression, respectively, of piano and bassoon.

Now that we have defined a standard mode, we can propose other modes of encoding a texture by considering some local deviations of the standard mode according to a given criterion. In this paper, we propose other four modes that do not necessarily constitute an exhaustive taxonomy since other modes can be presented according to other perspectives for decoding a texture into a

[49] Note that although the pair $[1,3]$ and [3] (m. 8) could be also executed by the piano, I decided to vary it to gradually prepare the entrance of the solo flute (m. 9).

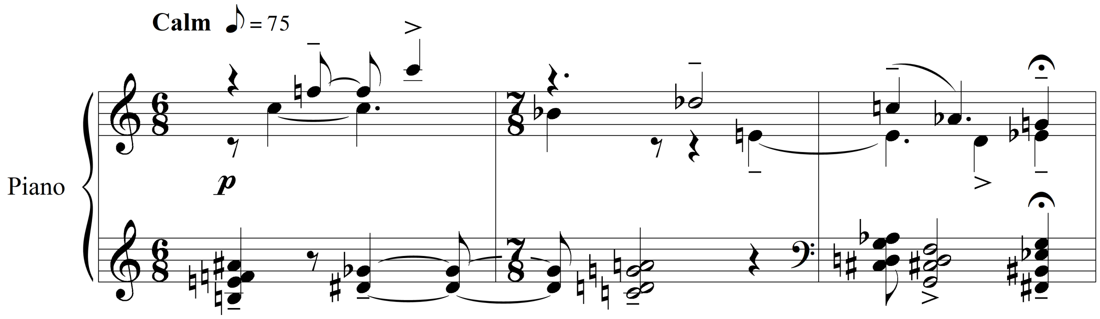


Figure 15: *Partitional complex {$1^2 4$} decoded into the score.*

musical score: *partitional complex*, *evolving realization*, *colorization*, and *montage*. Each one of these modes is related to a musical domain involved during the creative process. Note that a deeper elaboration of these modes, as well as the inclusion of examples of their effect, is out of the scope of this paper since our goal here is to initiate the discussion on how texture can be decoded in the musical score.

Partitional complex is an elaboration of a standard partition, proposed by Gentil-Nunes and Ramos (see [12] and [8]), to include local deviations from the referential partition. In some cases, rests in the parts introduce subsets of the partition, called *subpartitions*, and in other cases the changes in differentiation criterion between the parts produces merged parts, called *subsums*. Let parts $\{a, b, c\}$ be the parts of a partition K. The partitional complex enables that a textural sequence may include any of the six subpartitions of K ($\{a\}$,$\{b\}$,$\{c\}$,$\{a, b\}$,$\{a, c\}$, and $\{b, c\}$), or any of its four subsums ($\{a + b\}$,$\{b + c\}$,$\{a + c\}$, and $\{a + b + c\}$), or the combinations of subpartitions and subsums, called *subpartitions of subsums* ($\{a,(b + c)\}$,$\{b,(a + c)\}$,and $\{c,(a + b)\}$).

All subpartition, subsum, and subpartitions of subums are subordinate to partitional reference K, forming a hierarchical organization where textural changes are understood as oscillations within K. Obviously, the window of observation of supartition, subsum, and subpartitions of subums is minimal and necessarily shorter than the window of observation of the whole partitional complex K. For example, the partitional complex $\{1^2 4\}$[50] has four different partitions: the referential partition itself ([$1^2 4$]), two subpartitions ([1] and [4]), three subsums ([2], [5], and [6]), and two subpartitions of subsums ([2, 4] and [1, 5]).[51]. The use of this partitional complex is shown in Figure 15. Note that according to standard mode each supartition, subsum, and subpartitions of subums would be interpreted as a different referential partition instead of the unfold of the referential partition $\{1^2 4\}$ within a partitional complex.

Evolving realization concerns the presentation of blocks in the score under a temporal perspective. In addition to standard mode, where a block is formed by vertical alignment of its threads according to a given collaborational criterion, evolving realization may consider the potential of a block to be temporally constructed by successive superposition of its constituting parts. Thus, the block is articulated through the cumulative sustained sound of threads as an outcome of their static motion.[52] In evolving realization, the block is the referential partition (or one of its parts, in the case of a block within a partition with multiple parts). Note that the block is not understood as block until it is fully constructed; or one can say it is retrospectively referential. Thus, evolving realization demands a wider window of observation to consider the entire unfolding of the block

[50] We note a referential partition of a complex within "{ }" to differ from a regular partition.

[51] Of course, we are not considering the redundancies caused by part {1}.

[52] This procedure is known as an orchestration technique called *pyramid construction*.

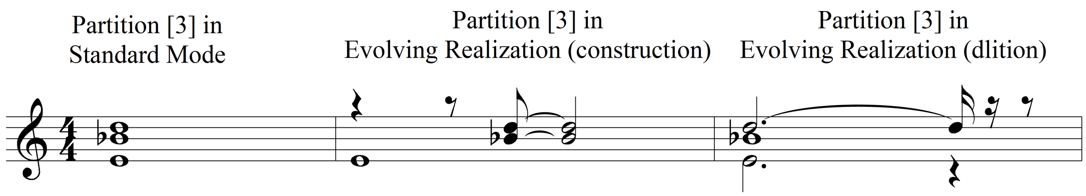


Figure 16: *Three ways to decode partition [3]: 1) standard mode, 2) evolving realization (construction), and 3) evolving realization (dilution).*

and therefore it is the most abstract mode of decoding a texture.

Through evolving realization any block can be accomplished from the cumulative superposition of the partitions of its cardinality. Given a referential partition [3], for example, while standard realization and partitional complex would present it as a simultaneous [3], evolving realization would include other two possible partitions by the superpositions of the threads: $[1, 2]$ and $[1^3]$. In this case, both partitions are understood as an unfold of referential partition [3]. Another possible application of evolving realization concerns the opposite sense, in which a simultaneous block is "diluted" by gradually removing their component parts. Figure 16 provides three examples of decoding partition [3] in both standard mode and evolving realization. In standard mode, [3] is decoded according to the note onset of the threads. All threads are simultaneously articulated. For evolving realization, first the block is constructed by the superposition of threads in a static motion, then the block is diluted by excluding the its threads. Note that if we analyze these two evolving examples through standard mode, the sequence would be interpreted as, respectively, the partition chain $<[1][1^2]>$ and $<[1^3][1^2][1]>$ since, according to the onset note, the threads are in contrapositional relation throughout the whole sequence. However, in evolving realization, both sequences are understood as a temporal elaboration of the referential partition [3] considering the motion criterion to assemble the block.

Other parameters such as timbre and dynamics, in addition to pitch and time, also may affect the decoding of a referential partition. This may occur in two ways: 1) a referential partition may be defined by timbre differences including octave doublings; or 2) differences that are noted in the score may give the impression that a referential partition is changing when it is not. Both cases correspond to what we call colorization mode, an encoding process that has much to do with orchestration and the way in which scores are written. Thus, colorization can therefore be interpreted either with smaller and/or wider windows of observation. A small window shows the doubling of notes within partitions (case 1 above). A wider window of observation can reveal relationships among different partitions within a passage (case 1 or 2 above). For example, in Figure 17, despite the fact that the score presents an elaborated texture with the combination of multiple parts, according to colorization, the whole sequence is understood as a creative unfolding of a single line (partition [1]) as a way to embellish this referential partition through unisons, doublings, and superposition.

Finally, the montage mode considers the various ways to connect two adjacent textures. After decoding them, the composer may decide how they can be linearly presented, considering three different cases: 1) simple juxtaposition of both, with no overlapping and possibly a rest between the two partitions; 2) overlapping or elision of the two textures or any of their parts (considering a wider window of observation); and 3) juxtaposition with *trigger parts* connecting them. Trigger parts are parts or partitions that are added to the point of juxtaposition of two partitions to highlight the change of one partition to the next. Triggers can either lead to the point

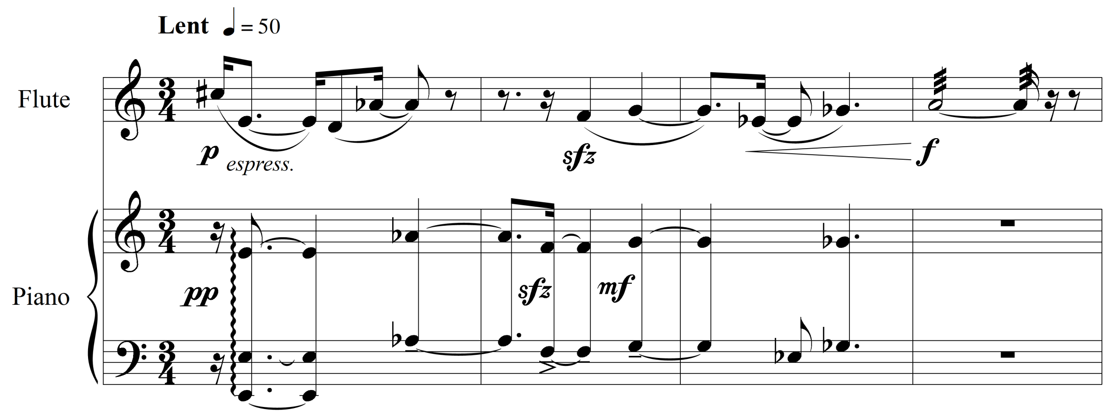


Figure 17: *Colorization mode applied to decode the referential partition [1] into the score.*

of juxtaposition (producing the effect of an upbeat), or they can occur at the point of juxtaposition (producing the effect of a downbeat). In the case of a simple juxtaposition (case 1 above), each texture is understood individually without affect their presentation in the score, but in the last two cases presented above, we may ignore the texture produced by either their overlapping or by the inclusion of a trigger part. For example, in the sequence of referential partitions <[1, 2][4][1]> (Figure 18) the rising gesture of the flute (partition [1] in m. 2) from partitions [1, 2] to [4], rather than configurating a new part to change partition [1, 2] to [$1^2$2], consists of a trigger part to connect both partition with an upbeat effect. In this case, the part has an ornamental effect. Note that the pauses in partition [1, 2] are understood as an articulation under the partitional complex. Moreover, the line [1] of the bassoon that starts at the end of the block [4] (m. 3) is understood as an overlapping of partitions [4] and [1].

Figure 18: *Montage realization decoding the partition chain <[1, 2][4][1]> into the score.*

IV. Conclusion

The formalization of the textural spaces can constitute a refined tool for textural analysis and composition. Each space presents a specific level of details in the process of textural encoding, which is directly related to the degree of liberty a texture can provide within the compositional process. As an analytical tool, textural spaces, rather than just depicting the way the threads of sounds are organized into textural parts (with more or fewer details), provide a transformational perspective through the employment of operations, allowing us to comprehend how a texture is connected to another. Moreover, they can be used as a pedagogic tool for training aural perception of textures with different level of details.

A discussion on some creative potential of the spaces has been provided as a way for helping to fulfill the lack of a systematic compositional approach concerning organization of textures. In fact, the preliminary discussion of the creative application of textural spaces requires a further study in order to understand their potential within a composition to adjust the concepts and to improve their tools.

In this sense, textural spaces constitutes the first step toward the formalization of a *textural design*, an adaptation of the Robert Morris's *Compositional Designs* [25], in which pitches of a composition are pre-defined and organized into an abstract structure that is meant to be implemented as a composition according to the composer's aims. Thus, by using an abstract structure (matrix, array, graphs, and the like) of two or more dimensions that may contain textural classes, partitions, and/or partition layouts, may allow a composer to better understand his/her own creative process, enabling the discussion of the compositional process itself from a textural perspective. The formalization of such proposal is in course.

Finally, the definition of a standard mode of realizing a texture allows us to compare the possible deviation of a given texture in a more sophisticated and creative way. This can provide a more substantial information about the textures, which may impact the criteria for selecting them taking into account the various ways to decode them along a compositional work. Moreover, the formalization of evolving, colorization, and montage modes can provide new perspectives for the study of orchestration through a textural approach [20]. A further formalization of all modes of textural realization, their possible impacts in the creative process, and their potential of combination within a composition, is also in course.

References

[1] Alves, J. (2005). *Invariâncias e disposições texturais: do planejamento composicional à reflexão sobre o processo criativo.* Dissertation (Ph.D. in Music). Campinas: UNICAMP.

[2] Andrews, G. (1984). *The theory of partitions*. Cambridge: Cambridge University.

[3] Berry, W. (1976). *Structural Functions in Music*. New York: Dover.

[4] Codeço, A. (2014). *Gesto textural e planejamento composicional*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[5] Engen, M.; Vatter, V. (2017). On the dimension of downsets of integer partitions and compositions. *ArXiv E-prints*. Cornell University: arXiv:1703.06960[math.CO], March, pp.1-12.

[6] Forte, A. (1973). *The structure of atonal music*. New Haven: Yale University.

[7] Fortes, R. M. (2016). *Modelagem e particionamento de Unidades Musicais Sistêmicas*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[8] Gentil-Nunes, P. (2018). Nestings and Intersections between Partitional Complexes. *MusMat*: Brazilian Journal of Music and Mathematics. November 2017, I/2, pp.93-108.

[9] Gentil-Nunes, P. (2017). Partitiogram, Mnet, Vnet and Tnet: Embedded Abstractions Inside Compositional Games. In: *The Musical-Mathematical Mind*: Patterns and Transformations. Pareyon G., Pina-Romero S., Agustín-Aquino O., Lluis-Puebla E. (eds). Cham: Springer Verlag, pp.111-118.

[10] Gentil-Nunes, P. (2009). *Analise Particional: uma mediação entre composição e a Teoria das Partições*. Thesis (Ph.D. in Music). Centro de Letras e Artes da Universidade Federal do Estado do Rio de Janeiro.

[11] Gentil-Nunes, P. (2006). Parsemas e o método de Fux. In: *Revista Pesquisa e Música*. Rio de Janeiro: Conservatório Brasileiro de Música, I, pp.38-47.

[12] Gentil-Nunes, P.; Ramos, B. (2016). Complexos particionais. I Congresso Nacional de Música e Matemática da UFRJ.*Proceedings...* Rio de Janeiro: UFRJ, pp.117-126.

[13] Gentil-Nunes, P.; Carvalho, A. (2003). Densidade e linearidade na configuração de texturas musicais. IV Colóquio de Pesquisa do Programa de Pós-Graduação em Música da UFRJ. *Proceedings...*, Rio de Janeiro: UFRJ.

[14] Giusti, C.; Salvatore, P.; Sinha, D. (2012). The mod-2 Cohomology Rings of Symmetric Groups. *Journal of Topology*, 5/1, pp.169–198.

[15] Guigue, D. (2011). *Estética da sonoridade*. São Paulo: Perspectiva.

[16] Lucas, M. (1995). *Textura na música do século XX*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[17] Monteiro, F. (2014). *A imagem textura e timbre no Prélude à l'après-midi d'un faune sob a ótica do poema homônimo de Stéphane Mallarmé*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[18] Moreira, D. (2017). Textural Contour: A Proposal for Textural Hierarchy Through the Ranking of Partitions lexset. In: *The Musical-Mathematical Mind*: Patterns and Transformations. Pareyon G., Pina-Romero S., Agustín-Aquino O., Lluis-Puebla E. (eds). Cham: Springer Verlag, pp. 199-206.

[19] Moreira, D. (2016a). Contornos musicais e textura: perspectivas para análise e composição. IV Brazilian Symposium of Graduate Studies in Music (SIMPOM), Rio de Janeiro, Brazil.*Proceedings...*, Rio de Janeiro: UNIRIO, pp.99-109.

[20] Moreira, D. (2016b). Relações entre textura e orquestração a partir da análise da Promenade da suite Quadros em Exposição de Mussorgsky. PPGM/UFRJ Research Colloquium, 14, *Proceedings...*, Rio de Janeiro: UFRJ, pp.46-61.

[21] Moreira, D. (2015). *Perspectivas para a análise textural a partir da mediação entre a Teoria dos Contornos e a Análise Particional* Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[22] Moreira, D.; GENTIL-NUNES, P. (2016). Planejamento composicional no domínio da textura a partir da expansão dos espaços musicais. Congress of National Association for Research and Graduate Studies in Music (ANPPOM), 26, *Proceedings...*, Belo Horizonte: UEMG

[23] Moreira, D.; GENTIL-NUNES, P. (2015). Ornamental Functions in Textural Domain from the Proposal for a Textural Contour. Congress of National Association for Research and Graduate Studies in Music (ANPPOM), 25, *Proceedings...*, Vitória: UFES.

[24] Morris, R. (1995). Compositional Spaces and Other Territories. *Perspectives of in New Music*, 33/1-2, pp.328–358.

[25] Morris, R. (1987). *Composition with Pitch-classes*: A Theory of Compositional Design. Yale University Press.

[26] Mountain, R. (s.d) *Periodicity and Musical Texture*. Available in: http://armchair-researcher.com/Rooms/Research/Rooms/writings/articles/PeriodicityMusical-Texture.pdf, accessed on 05/18/2019.

[27] Ramos, B. (2017). *Análise de textura violonística: teoria e aplicação*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[28] Rinzler, P. (1999). *Jazz Arranging and Performance Practice*: A Guide for Small Ensembles. Lanham: Scarecrow Press.

[29] Santos, J. (2014). *A textura musical na obra de Pierre Boulez*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Universidade Federal do Rio de Janeiro.

[30] Schubert, A. (1999). *"Aura": uma análise textural*. Dissertation (Masters in Music). Programa de Pós-Graduação em Música, Centro de Letras e Artes, Escola de Música, Rio de Janeiro, Universidade Federal do Rio de Janeiro.

[31] Senna, C. (2007). *Textura musical*: forma e metáfora. Thesis (Ph.D. in Music). Centro de Letras e Artes, Rio de Janeiro, Universidade Federal do Estado do Rio de Janeiro.

[32] Shannon, C. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27/3, pp.379-423.

Computational Implementation of Henry Pousseur's Harmonic Networks Applied to Live-electronic Music

*Rael Bertarelli Gimenes Toffolo
State University of Maringá (UEM)
rbgtoffolo@uem.br

Abstract: *Considering the recent discussion involving the improvisational techniques and the more speculative structural process in the creation of live electronic music, we present, in this paper, the computational implementation of Pousseur's Harmonic Network in SuperCollider computer language in order to contribute to the development of real-time music. Initially, we present a brief review of the Pousseurian Harmonic thinking and how the Belgian composer created Harmonic Networks, discussing their uses for music making. Then, we present how Pousseur's Harmonic Network was implemented in SuperCollider computer language considering the algorithmic solutions to this task. Finally, we demonstrate how Harmonic Networks can be used in live electronics compositions.*
Keywords*: Pousseur's Harmonic Network. Live-electronics. SuperCollider.*

I. Introduction

In recent years, a great number of composers have invested in producing live-electronic works, and, consequently, there has been an increase in number of both theoretical and esthetical studies approaching such musical practice ([19, 18, 11]). It is thus only natural that favorable or opposing arguments to live electronic music spring in such context, and, even among those who favor such practice, there are, at times, antagonistic esthetical positions.

Dias ([2]), in his "Quarrel of times: a study on the aesthetic divergences in electroacoustic music", touches on one of the most popular discussions aroused at the beginning of live electronic music, which is that which refers to the interpretive temporal malleability resulting from the live electronic music practices that are potentially not possible to be reached in differed time electroacoustic music. For Dias, live electronic musicians have argued that such practice respects the historical continuity, leaving it to the performer the interpretive decisions, which were considerably limited in differed time electroacoustic music. On the other hand, supporters of music on differed time state that the tools available for the creation of live electronic music do not allow for such level of speculative control in relation to the creation and manipulation of the electroacoustic materials.

*I would like to tank the Research and Sound Production Laboratory (LAPPSO) at the State University of Maringá, and the friend and Computer Scientist Flávio Schiavoni.

Both arguments are debatable, as pointed out by the author. In the first case, the temporal malleability provided by live electronic music is a compositional decision. Its presence or absence is dependent on the plan and structure as conceived by the composer, and by the manner with which he/she manages such structures, which are not necessarily conditioned to the fact that the electroacoustic processes are not live. In the second case, the tools available for recent electroacoustic music production, both for differed time and live, are basically the same. The technological differences that used to permeate the synthesis and processing of the signal for differed time and live electronic music are almost inexistent today. Software such as Max/MSP, SuperCollider, Csound, PureData, among others, are used in both cases.

Not only has the recent technological development equated the techniques in both fields, but it has also contributed to the development of new interactional patterns, which are widespread over all the different forms of art production. Visual and performing artists are now committed to the development of interactive work, using a great deal of the tools once dedicated to music production, such as PureData, SuperCollider e Max/MSP.

In the visual arts, the concept of interactivity is usually discussed within the area of performance and it involves both the relationship between the spectator and the work, and the interaction between the different artistic languages, such as visuals and the body, sound and the body, sound and visuals, among many other possible combinations. Such speculations, resulted from the many different artistic languages, have fostered a great development among the most varied types of detectors and sensors that, by its turn, have significantly broadened the possibilities of the interaction man-machine.[1]

Simon Emmerson ([4]) dedicates his book "Living electronic music" to the study of the live electronic music status amid the recent technological developments, presenting an overview about the music esthetics in such context. He starts from the idea of the reintegration of the living or of the presence of the living[2] within contemporary artistic production, strongly based on the post-Pontyan philosophy[3], considering that the reinsertion of references in concrete music marks the beginning of this position. In this sense, he approaches the issue of how information about the dimension of the living can be extracted from the sound flow. Initially, he ponders how the many sensors are used to get information from the instrumentalists. Later, he dedicates to the concept of sonification, used to describe pieces that aim at creating musical structures from data collected from non-musical sources. For the author, such strand has its origins in the work of the theoretician and composer Iannis Xenakis, who used mathematical and statistical models for the creation of music structures. Such paradigm develops in such a way as to consider data extracted from biological or social structures, among others. Finally, he considers how the body has been the focus of attention from artists and musicians as source of information for art creation.

Within this broad overview and a certain technological fetishism associated to the strong proliferation of new means of production, as stated by Menezes ([12, p.9])[4], several works have

[1] For an overview on the relationship man/machine, in the music field, check ([20, 21, 7]). For a view on the uses of technology in visual and performing arts, check Domingues ([3]).

[2] This living dimension considered by Emmerson encompasses from the electroacoustic music instrumentalist to the extraction of information from nature, populational dynamics from several species, changes in climate patterns from any biological sphere and others.

[3] The author presents a more direct relationship to philosophers following the Merleau-Ponty phenomenology in his text, such as Varela ([23]), Maturana ([10]), among others. For an overview on philosophers of such strand, check Petitot ([13]).

[4] Menezes' words: "And so it was, for today young composers have much more access to spectral manipulations that will enable him to relate to the new technologies than the corporate and highly bureaucratic that are the basis to orchestral music. [. . .] Such fact is, however, an unquestionable confirmation: even (and above all) institutional recognition of the supremacy of the electroacoustic music is needed. However, full awareness of the technical and aesthetical potentialities of the new music compositional tools is fundamental, watching constantly to the risks of an unreasonable fetishism in

emerged, giving origin to a new quarrel, not really related to music in differed or real-time, but rather related to the opposition between improvised versus pre-elaborated music, with a higher or lower structural degree.

Philippe Manoury, one of the pioneers of real-time electroacoustic composition, shares his impressions on the abundance of both improvised and performed works:

> I'm forced to observe that musicians who come close to real-time music in a decisive way, it is not within my close esthetical family— that of composers—that I have found a more committed engagement, but in an esthetical trend that lies way far from my artistic orientation: that of improvised music and 'performers'. Such curious situation has driven me to isolation for quite a long time, for such overlap of esthetical and technological orientations that I considered my own, was only rarely shared by others. ([9, p. 6])[5]

We can then realize that improvisation is strongly present in real-time electroacoustic music ([5, 6, 22, 1]). In this context, a new compositional strand has been developed, the so called live-coding ([8]), which consists of the creation of programming codes in a collective form and in real-time.

Alongside this situation, it seems that the quantity of interconnections between the several dimensions of the artistic trade, occurred amid the exchange of information obtained by sensors of all kinds, have made space for a considerably broad universe of creative possibilities, pointing to new issues such as: how to create a relationship between data obtained by sensors, which are placed in a specific symbolic field, and the computational codes that will generate a piece of audio, a graphic animation, or will control one of the parameters of an audio processing coming from an instrumentalist?

It is within this context that this research is placed. How to foster the creation of works built in real-time, one that benefit from the temporal and interpretive malleability that real-time music allows for, keeping in mind the special attention to the speculation of structures and materials?[6]

It is for this reason that this work is out to benefit from the harmonic speculations coming from the Belgian composer Henri Pousseur and apply them to real-time music. As we will see along the lines in this text, Pousseur's harmonic networks are characterized as an interesting relationship tool between several harmonic systems from different historical periods. In this sense, Pousseur's harmonic network technique arises as an interesting field of study for contemporary art.

II. Pousseur in context

> How to "rhyme" a citation of Gluck or Monteverdi with one of Webern in a single composition (two grammatical domains that have so far seemed the exact opposite and incompatible); how to "conjugate" them, find common functions, and, to start, establish among them a series of intermediate types capable of convincing the musical ear that they once pertained to a more general common domain? ([15, p. 194])

relation to the technological means, trend which is constantly trying to be established in the capitalist cultural industry in a nefarious way."

[5] "Force m'est de constater que, parmi tous les musiciens qui se sont approchés du temps réel de façon décisive, ce n'est pas dans ma famille esthétique proche – celle des compositeurs – que j'ai trouvé l'engagement le plus conséquent, mais dans un courant esthétique beaucoup plus éloigné de mes orientations artistiques : celui des musiques improvisées et des 'performers´". Cette curieuse situation m'a laissé assez isolé pendant longtemps, car cette union d'orientations esthétique et technologique qui était la mienne, n'était que rarement partagée par d'autres. This and the next translations were made by the present author.

[6] Our proposition does not aspire to criticize or invalidate the improvisational procedures but only aim to demonstrate other process to creating live-electronic music.

Let's consider the intention expressed by Pousseur, in his "Rameau's apotheosis – an essay on the harmonic question", briefly, but not superficially, and the developments of the Pousseurian theory as it sets out to reach a meta-grammar capable of making "rhymes" from both universes, or more precisely, harmonic spaces — as he will later denominate.

It is worth mentioning that there is no naive rescue of the tonal functions in the composer's proposition, ignoring all the musical development reached by the post-Weberian generation, of which, by the way, Pousseur himself is part, mainly when expressing the will to reconsider the Monteverdian universe within the current composition. As he alerts:

> In effect, having in mind the high degree of elaboration and the extreme hegemony that this dimension exercises in tonal music, as well as the considerable cultural weigh present in the harmonic perception to Occidental ears, it is in such frame that there seems to lie the greatest dangers of a purely reactionary evolution, of a throwback that we evidently cannot admit. ([15, p. 192]).

Back to Pousseur's central issue, the one in question here: the ways of making rhymes from the tonal universe expressed symbolically by Monteverdi's music and Webern's multi-polar universe. Such desire emerges from a deep analysis of the music of his time, in special, Integral Serialism. For the author, serialism, which provided for such a rich development of the compositional practice, needed a deep review, mainly permeated by considering perceptual and even structural issues. Besides, as we will be able to confirm later, Pousseur sought different ways to redeem functional, formal, and perceptual aspects that seemed to have been neglected by strict serialism. He was obviously not the only one to make efforts in this sense, and even the Concept of Groups, back at the beginning of the Integral Serialism, had aimed at redeeming certain perceptual logics not considered by the pointillist serialism of the first phase, as he states:

> This evolution started very carefully, and, if we can recognize its first symptoms at the beginning of the so called "group techniques" (from Le Marteau sans Maître by Boulez, for example), if we can find more established demonstrations in works of the final 50s, as in Gruppen by Stockhausen or in Circles by Berio, it is quite evident that it is about the harmonic plan that major concerns were demonstrated ([15, p. 191]).

For Pousseur, serialism was lacking in its structural affiliation, reductionist in a way, to the pitch parameter as the primary element. Such abstraction process raised questions about certain perceptual properties, so dear to the composer, demonstrating a strong affiliation of his to a phenomenology of listening.

In this sense, all the typical structure processes present in integral serialism are set up in a set of rules that promote a certain paralysis or perceptual homogenization, which function as to stop properties or characteristics of music from the past from emerging, more specifically, the tonal function. If, on the one hand, this process was extremely positive in providing for a whole new universe of possibilities, it led, on the other hand, to a homogenization to be viewed as worrying for Pousseur. As he states:

> The most evident consequence of this partially voluntary paralysis is the extraordinary distance (. . .) between the level of constructive intentions and the perceptive results. The structural work, from which traditional serial techniques are the most notable example (permutation, etc), is produced on an extremely abstract plan, on which relationships to a very high intellect coefficient are made, at the same time that the perceptive result is situated, on the contrary, on the most concrete plan possible, on which pitch is nothing more than one aspect, of many, of timbre, of the more immediate

> materiality of sound. The connection between these two plans is deliberately lacking in intermediate solid support, which could be, nonetheless, provided by both a realistic as well as a logical reflection ([15, p. 173]).

From then on, Pousseur starts to investigate the parameter of pitch as an attempt to redeem certain aspects that seem to have been abandoned amidst the high degree of abstraction of the serial engendering. For Pousseur, such rescue will demonstrate that the perception of the frequencies will enable several levels of meaning, pre-musical at times, on which new musical constructions can emerge, considering its uses as color (that could generate musical meaning due to its metaphorical or symbolic properties), melody (that result from the perception of scalar relations between pitch in a temporal stream), harmony (considering its parental, proportion, attraction, and repulsion relationships), and combinatorial (considering the more abstract uses originated from permutations and inversions).

Alongside his worries on the pitch functions, Pousseur also was highly fascinated for the game of literary citations developed by the writer Michel Butor, his partner in his *Votre Faust*, who would feel more accomplished in terms of his compositions doing something similar in music. He thus aimed at finding an organic way to bring together several citations from music history within his compositional language. The key to the process would be making such set of citations "rhyme" in a way to incorporate to Pousseur's musical discourse, not as "strange bodies", as collages, but in such a way to accommodate his language so that the citations would fall into place "naturally".

> [...] I thought I would only be satisfied with my musical language the day I feel I'm able to insert old elements in it in such an organic way as Bach would integrate the Protestant chorale ([15, p. 193]).

Therefore, Pousseur was not interested in including only citations by other composers or historical periods, but to put together different ways to characterize the harmonic spaces in which such musical contexts would develop, and, once such "background structures" were detected, find a musical meta-grammar that relate them:

> An initial possibility would consist in taking an example from music history and find the first elements of a "grammatical scale", of an integral harmonic space in the evolution that conducted tonality to atonality (especially in Viennese music), with the gradual logics known to us. [...] It was then necessary to find an organizational system at the same time ample and coherent enough to fit all historical cases in one place, in a way that they would finally seem to have been engendered exactly by this system ([15, p. 194]).

In this sense, we notice that, in Pousseur's thoughts, there is a deep complementarity between the search for the organic use of citations and harmonic speculations that are, on their turn, a result from his revisionist criticism to serialism. Such complementarity is expressed by the composer first when considering the harmonic aspects that instigated him:

> (...) I was not convinced that I had lost "deepness" definitely that only the harmonic functions – tonal (or post-tonal) or modal, occidental or extra-European, known or only sensed – seemed to be able to offer, and that I saw it as one of the most precious riches, one of the central values of music as a whole, susceptible to qualify it in its irreducible specificity [...] ([15, p. 192]).

Still on the same text, when considering the historical referencing that encompasses both a citational and a functional level, the author states that when he "heard a symphony by Monteverdi or a Lied by Schumann, a Hindi Raga or a symphonic work by Debussy, he could not help it but to think, with certain nostalgia, that we voluntarily deprived ourselves of something irreplaceable, and that I would do unquestionably better if I did not let myself be deprived of that" ([15, p. 192]).

It is within such context that includes certain criticism to serialism, a need to redeem formal aspects from music history and a solid urge for the citation organically inserted in his musical language that Pousseur starts to investigate ways in which to put together a meta-grammar that would relate all these concepts.

III. Pousseur's steps for developing the Harmonic Networks

Pousseur's first step was to try to build a method with which he would be able to calculate the polarizing forces of the musical intervals due to his belief that even in Webern music one can find the same principles of tonal harmony polarization, differing only in relation to its vectors: while in tonal music such polarizations were used in a convergent or centrifugal manner, in Webern music we will find a game of centripetal or multi-polar polarizations. These polarization studies are too similar to the Edmond Costère theories. Pousseur arranged the intervals in a progressive sequence from more consonants to more dissonants, therefore more polars or apolars, as we can see in Figure 1.

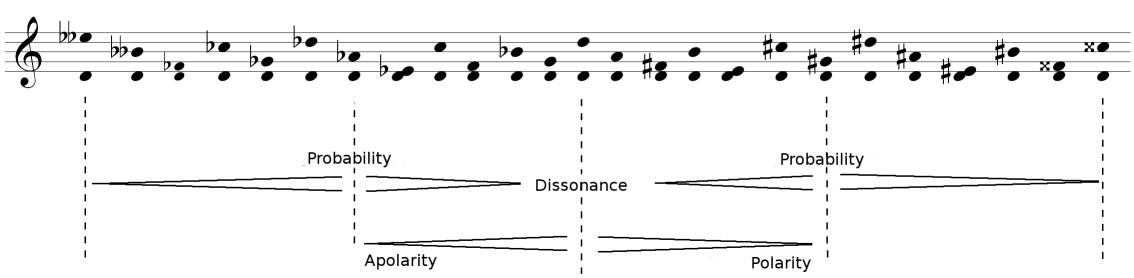


Figure 1: *Interval Polarity Table, adapted from Pousseur, 2009.*

From the table in Figure 1, Pousseur creates a chart of three-sound chords that are set according to their sequence. Still, Pousseur considers that the handling of data deriving from this form of classification would be quite time-consuming for the engendering of compositional materials, especially when expanding the chart to sets of four or more sounds is needed.

He then exposes another method that would become known as Cyclic Permutations, which consist of a sequential transformation of a series of original notes derived from the alteration of some of their elements by a continuous transposition process to a previously selected interval. If a series of notes formed by the chromatic total is elected, keeping some characteristic notes frozen and transposing others step by step to a previously selected interval, a perfect fourth, for example, some interesting transformations occur, as we can see in Figure 2.

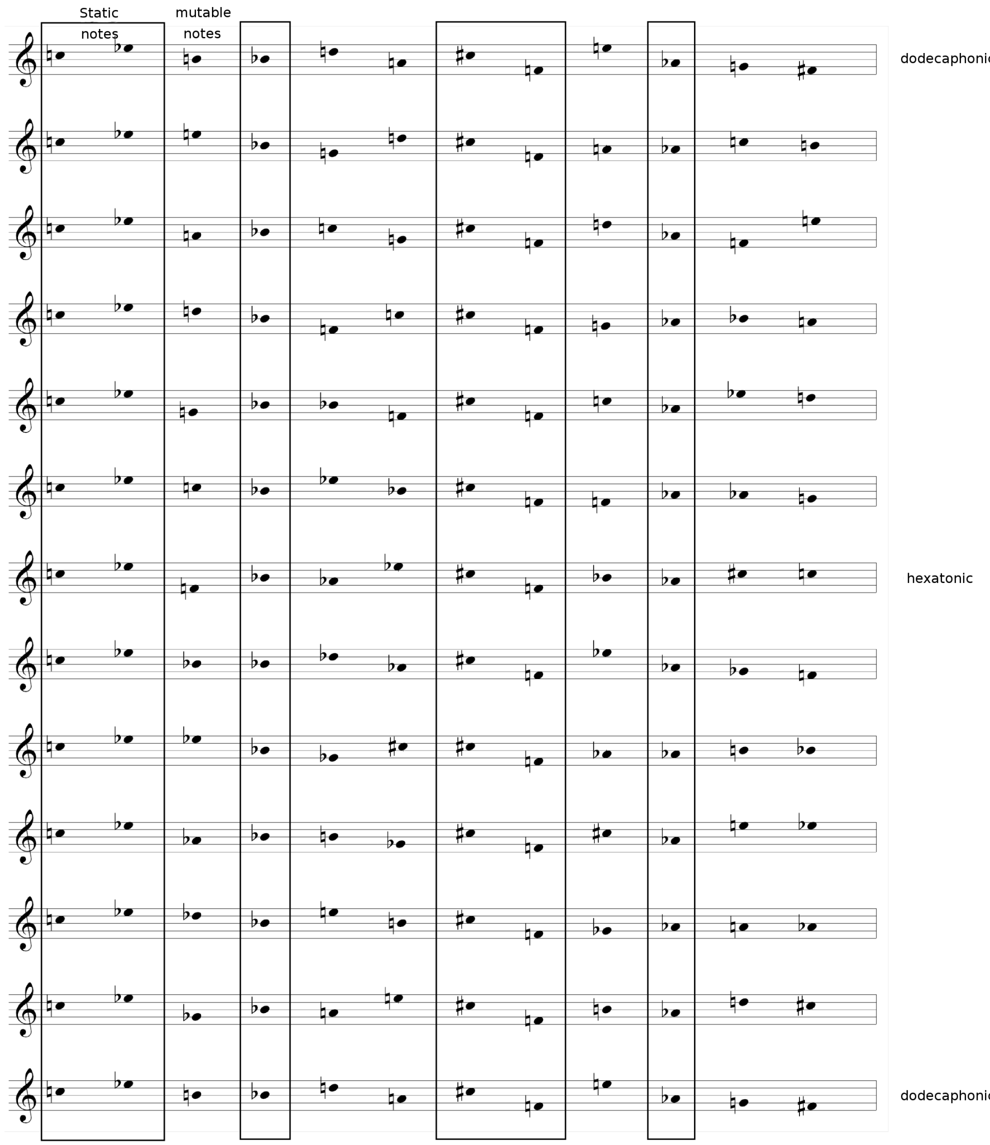


Figure 2: *Cyclic Permutations process, adapted from Menezes (2002).*

It is observable that, when the sixth permutation is reached, a group of notes expresses a hexatonic set and then, at the end of the permutation process, is back at the original series. Besides the several properties originated as new notes appear in each permutation, Pousseur, with this process, is closer to his attempt to find a relationship system between different harmonic spaces, a process that is built with a gradual, directional passage, from a relational system to another, and, in this case, from a chromatic set to a reduced six-note set (hexatonic).

From these two processes, Pousseur proposes a relationship form that will contribute to his harmonic networks. By setting interval sequences of the same size on an horizontal and a vertical axis and make transpositions at another interval fixed rate, in the fashion of the cyclic permutations, Pousseur was able to verify the emergence of several relationship properties or certain "harmonic fields" that result in true relationship networks. Here is, thus, the prototype of a harmonic network, still two-dimensional that was later defined:

> [. . .] a network, as understood here, is a distribution of notes (to be specified later as to what they represent) formed by several axes (starting with two) that are each characterized as a chain of a single interval.[7] ([14, p. 249]).

While characterizing this relationship space obtained through crossing both axes two-dimensionally, Pousseur notes that there is a harmonic logics that emerges from the distances between the notes that relate to each other in a closer or more distant fashion:

> We cannot forget, however, that the principle of the method itself resides in the wish to build nodes of all kinds so that the effective elementary musical relationships, therefore "in time", (analyzed or composed, melodic or harmonic) are the closest possible, expressing themselves especially among the network neighboring tones, in one direction or another.[8].

Such network pitch relationship structures are set up as both a compositional as well as analytical tool. In order to build such networks, certain intervals are fixed on each one of the axes and bring, each one of them, their polarizing, neutral or non-polar characteristic. From this process, a complex net emerges, or in other words, a relationship network depending on the distance that it is found from the center of net, that is, depending on its neighboring level. Several harmonic spaces can be represented, characterized or found in a two-dimensional network, as can be observed in several of Pousseur's example, in special in his 1998 text, "Applications analytiques de la 'Technique des réseaux"'. In this work, which complements his speculations on harmonic networks presented in Apoteose by Rameau, Pousseur demonstrates how networks can be used in musical analysis with examples from the works of Bartók, Debussy, and Webern. He also demonstrates their potentialities to express tonal and even modal spaces. In such context, Pousseur explains how a two-dimensional network can easily represent a modal relationship space and how the insertion of a third axis becomes effective for characterizing the tonal space, demonstrating that the inclusion of this third axis would be equivalent to the development of perspective in visual arts.

> But from the moment the polyphonic practice intensifies, despite the initial resistance posed by clergymen, who wanted thirds to continue to be used (therefore considered) as "dissonances" (pre-attractive) and that only eighths and fifths were recognized as perfect consonances, the acoustic experience (with its sound combinations exalted by the resonance of architecture at times) would impose itself: for about two centuries, thirds have reached their place (almost) equally prominent, and the triad system starts to settle. This revolution can be theoretically expressed by the junction, in the fundamental diatonic network, of a third dimension (which strangely happened at the same time as visual arts discovered depth: and that could hardly have been by chance!).[9]

[7][. . .] "un réseau, au sens ou l'entend ici, est une distribuition de notes (on précisera plus tard ce qu'elles représentent) selon plusieurs (pour comencer deux) axes qui se caractérisent chacun comme une chaîne d'un seul et même intervale."

[8]"Il ne faut toutefois pas oublier que le principe même de la méthode réside dans la volonté de construire le lacis de telle sorte que les relations musicales élémentaires effectives, donc 'en-temps', (analysées ou composées, mélodiques ou accordiques) soient les plus serrées possibles, s'expriment principalement entre notes voisines du réseau, dans un sens ou dans l'autre." ([14, p. 249])

[9]"Mais dès que la pratique polyphonique s'intensifia, et malgré la résistance opposée d'abord par les cléricature, qui voulait que les tierces continuent à être *utilisées* (parce que *considerées*) como des 'dissonances' (pré-atractives) et que

In his final tridimensional version, harmonic networks consist of axes on which sequences of notes from pre-determined intervals are set. Pousseur's initial proposal consists of a selection of axes that he considers the expression of intervals that are "fundamental of the aural acoustic space, not only in their exploration of tonality, but also in their more probable natural potential". In other words, he considers the relationships by thirds and fifths, as he proposed in his two-dimensional networks, but he includes the octave as a factor of great importance for the characterization of the tonal space. Therefore, the network would be composed by three axes that will be expressed tridimensionally here as X, Y, and Z. If we place the perfect fifth interval on the X axis, the perfect eighth on the Y, and the major third on the Z, we will obtain a relationship network as presented in Figure 3.

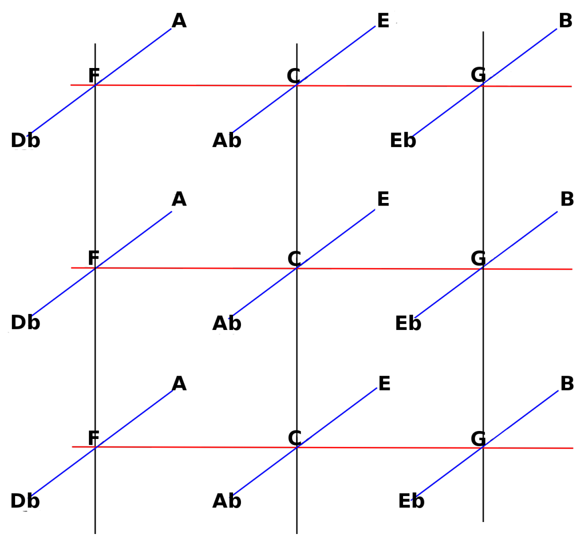


Figure 3: *Harmonic Network with P5 in X-axis, P8 in Y-axis, and M3 in Z-axisPerfect Fifth, Perfect Octave and Major Third.*

This way, we will have the possibility to represent any set of notes in a relational space. It is interesting to note that, depending on the choice of intervals for each one of the axes, we will reach a set of relationships that express a certain harmonic space, and even a theoretical system. In this initial network proposed by Pousseur, we will reach, in a very immediate manner, the representation of a tonal space. Let's take, for instance, the notes from the C major triad, building the network starting at C, we will be able to find such notes in the more central positions of the network. It is possible to infer, therefore, that similarly to Riemann ([16]) or Schoenberg ([**?**]), to mention a few, that present in their tonal functionality charts as regions closer or more distant, in Pousseur's networks we will also find the representation of the acoustic space both between the notes and between the harmony expressed by the distance in relation to the central axis of the network. And yet, interestingly enough, when the axes are modified, we will find other acoustic spaces available with their relationships also expressed in terms of their distance from the center.

seules les octaves et quintes soient reconnues comme consonances parfaites, l´expérience acoustique (avec ses sons de combinaison souvent exaltés par le résonance des architectures) s'imposa: au bout de quelque deux siècles, les tierces avaient acquis leur place (presque) également prééminente, et le système des triades avait commencé à se mettre en place. Cette révolution peut s'exprimer théoriquement par l'ajout, au réseau diatonique fondamental, d'une troisième dimension (qui suivent bient étrangement au moment même où les arts plastiques découvert également la profondeur: ce ne peut guère être dû au seul hasard!)" ([14, p. 255])

Besides the networks graphic representation, as in Figure 3, they can be represented in musical writing, as in Figure 4.

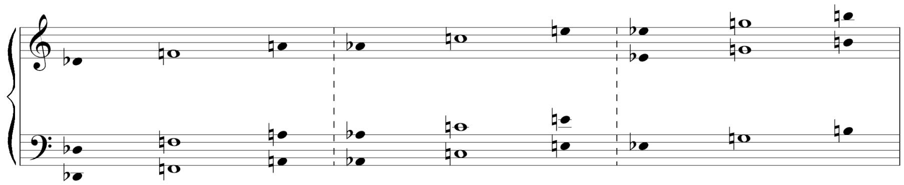


Figure 4: *Pousseur's Harmonic Network with axis: P8, P5 and M3, represented in music notation.*

We can notice in Figure 4 that the Y axis, which in Pousseur's original network is equivalent to the eighth interval, is represented vertically. Each one of the notes from this axis is "projected" in an infinite sequence of ascending and descending eighths. The X axis, equivalent to perfect fifths, is represented by the horizontal sequence of the white notes (F, C, G in the example), that is also projected indefinitely to both directions. Finally, the Z axis, equivalent to major thirds, is represented by the ascending and descending major thirds that are formed between the white and black notes. One must mentally project a virtual axis of thirds, in this example, that emerges diagonally and progresses, as on the other axes, indefinitely in both directions.

The basic operation proposed by Pousseur for the generation of material for composition consists of locating any intervallic set up, either chordal or melodic, in its structure. Figure 5 exemplifies the process of locating a musical excerpt (with numbered notes) and how such notes are located in the network structure.

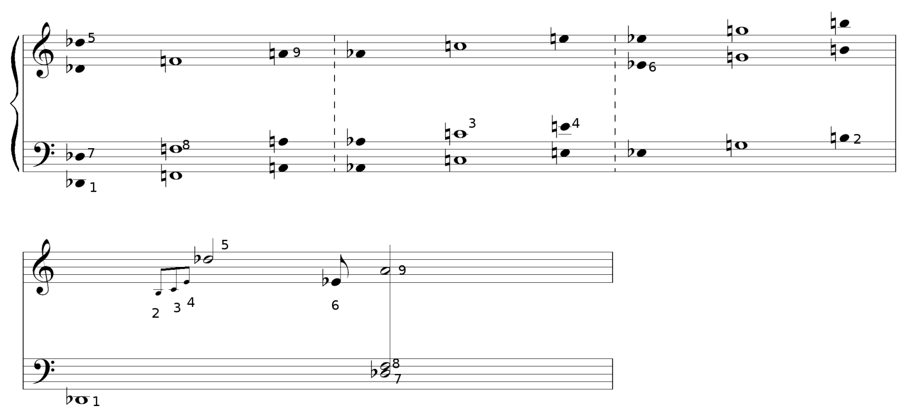


Figure 5: *Localization of a musical excerpt in a Original Harmonic Network.*

Once the notes from the musical excerpt in the reference network structure are "projected", the process of "deformation" of the network basic structure starts, and, by locating the same positions marked in the previous stage, the musical excerpt reconfigured by the deformation of the network axes will be revealed. As exemplified in Figure 6, we can observe the new musical excerpt obtained from a network on which the axes were modified from perfect fifth to augmented fifth, eighth to major seventh, and major third to minor third.

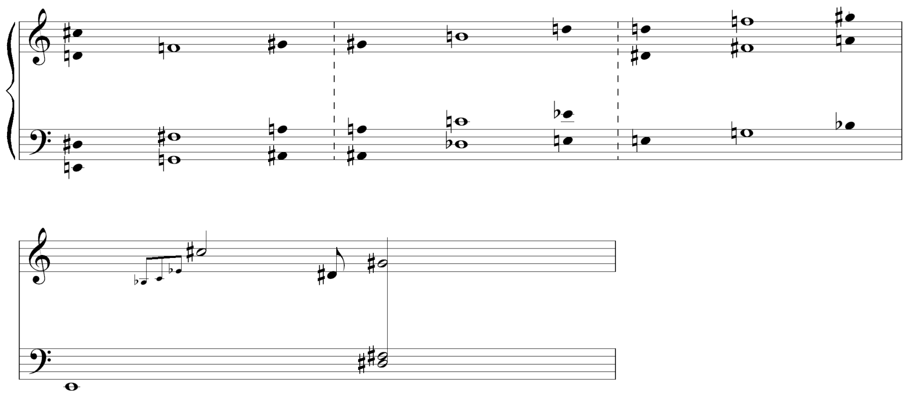


Figure 6: *Musical excerpt recovered from a "deformed" Harmonic Network. X-axis in aug4, Y-axis in M7 and Z-axis in m3.*

The possibility of changing the constitution of its axes and re-locate notes previously placed in an original network on a transformed network constitutes one of the most relevant contributions from network techniques to contemporary music. Here we finally find the relational tool that Pousseur looked for in order to rhyme Monteverdi and Webern. The networks are set up as the meta-grammar that connects the several harmonic spaces projecting a structure obtained from one of these spaces into another. When notes from a chord, or a highly tonal excerpt are located on an obviously tonal axis (5J, 8J, 3M), and, later, finding the same positions on another network with transformed axes, so that we will obtain a much less tonal acoustic space — as, for instance, a network of axes formed by the triton, major seventh, and minor second —, we will reach a collection of notes that are much closer to the non tonal pole, as Pousseur intended: a way to relate different harmonic universes from a single meta-grammar. Besides, the melodic patterns are, in general, kept proportionally steady, except on rare occasions, when transformations on a given axis are of a much higher caliber than the transformations on the other axes.

Such properties allow us to speculate how to implement the harmonic networks as computational algorithm in a way as to use such tool both for Algorithmic Composition, as well as to expand its functionalities to the use in electroacoustic works in real-time.

IV. Implementing the Pousseur's Harmonic Network in Supercollider Language

The harmonic network computational algorithm was implemented on the SuperCollider audio programming language, which is widely used by composers all over the world. Because it is an OpenSource application, it is readily available for free for all operational systems, (Macintosh, Windows, and Linux), what increases the portability of works and tools created in them considerably. SuperCollider ([24]) is both an environment for development and a programming language created and launched in 1996 by James MacCartney. It is a very powerful computational tool both for audio and video signal synthesis and processing in differed time, as well as for applications in real-time, presenting great functionalities for the algorithmic composition. One of the most interesting characteristics of such programming language is that it can be expanded, including new functionalities, from its own language. On other systems, as PureData or Max/MSP, the creation of new functionalities depends on a wide knowledge of programming languages (such as C or C++), software compilation. The disadvantage is that learning to work on SuperCollider is a little more time consuming than dealing with graphic systems such as PureData or Max/MSP, but not so different from understanding how to use languages such as Csound, for example.

One of the problems found in computationally implementing the networks[10] was the location of the chosen notes in its structure. Pousseur does not specify which procedure should be adopted, since the notes are recurring in distinct areas of the network, especially if we consider that the harmonic networks are endless in all directions of their axes. It is for this reason that we have chosen, and for practical purposes — and we do believe that Pousseur has done the same, since he considered that the neighborhood, therefore the shortest distances from the center of the network, is an important characteristic in representing the harmonic space in mind —, to always locate the desired notes in the closest position to the central axis. Therefore, the selected algorithm solves such dead-lock, as we shall see below.

i. The algorithm explained

The computational implementation of a problem from the real world depends on its representation. At first, the original network is an infinite space and, as such, cannot be represented computationally. For this reason, we chose to adopt the MIDI scale with 128 notes (from 0 to 127) to limit the network sample space.[11] The second issue to be considered was the selection of the best method to calculate the distances between the different spots on the network and, thus, create the sample space from the original network.

The creation of the original network depends on the entry of 4 parameters: central note and axes X, Y, and Z step sizes (intervals). For example: a network starting with values 60, 5, 8, and 12 is equivalent to a network with a middle C as its central note (60 on the MIDI representation) and axes X, Y, and Z equivalent to the perfect fourth (5 semitones), major sixth (8 semitones), and the perfect octave (12 semitones).

From the initialization and delimitation of the sample space scope (size), three data structures were used to create the initial network. The first consists of a list with the 128 possible notes, used to mark the notes that are already part of the network and the ones that still are not, as shown in Figure 7.

This way, we can go through the network and check which nodes are already part of the network and mark them on this data structure. The selected algorithm, as we will see in more detail below, starts from the network central axis and searches, step by step, each note that constitutes the network, going upwards and downwards on each of the axes, until it locates all the desired notes. When the algorithm locates one of the desired notes on a certain spot, it will not look for that note anymore. That means that, as the algorithm looks for notes starting from the network center, the first example of a certain note that is found by the algorithm will be the closest to the central axis.

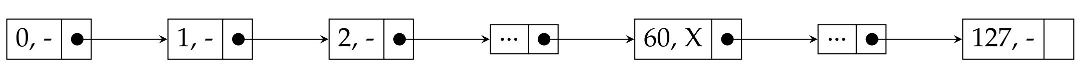


Figure 7: *List of visited nodes (X), and not visited nodes (-). Middle C (60 in MIDI scale) marked as visited.*

[10]The implementation of Pousseur's harmonic network counted on the invaluable help of the computer scientist Flávio Schiavoni who worked on this project suggesting very helpful algorithmic models.

[11]Sample space is a common term in computing that usually refers to the minimum and maximum limits of a set of data on a string, array, or even on the computer memory. What determines the sample space is of great importance in computational algorithm, especially when dealing with data that can be virtually infinite. The control of such space must be carefully taken care of so that execution errors are avoided, especially the ones originated from the overlap of information in the computer memory.

The second data structure used was a list of notes to be visited. As we find a note, it is added to this list, and informs that all the axes (ascending and descending intervals) will be visited next. Such list is a *fifo* (first in, first out) in which all new data added to the end of the list is first consumed.

As we simulate the behavior from the example above (60, 5, 8, 12), we put the central note in line and initialize the algorithm through its ascending perfect fourth, descending perfect fourth, ascending minor sixth, descending minor sixth, ascending perfect octave, and descending perfect octave. Each note that has not been visited will be placed in line so that its axes can be visited. At the end of the visit of the first note, since none of the axes has been visited yet, we will reach a line as the represented in Figure 8.

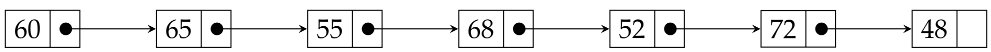


Figure 8: *Visited list in the first algorithm step. Values are added in order to visit: Starting note 60; X-axis ascending and descending (60 + 5; 60 - 5); Y-axis ascending and descending (60 + 8; 60 - 8); Z-axis ascending and descending (60 + 12; 60 - 12).*

At this moment, we mark the initial node as visited in the list of visited nodes (first data structure), removing it from the list of notes to be visited, and moving to the next element in line to be visited. The list of visited nodes, when we move to the second element, is represented in Figure 9.

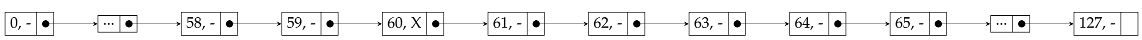


Figure 9: *List of visited nodes in the second step of algorithm.*

Note that only the central note (60) is marked as visited. That stops us from trying to visit it again.

On each step of the algorithm, the respective ascending and descending axes of each one of the notes on this list will be included in the list to be visited. In the case of the note 65 (ascending perfect fourth from the initial note), the values as illustrated in Figure 10 will be included.

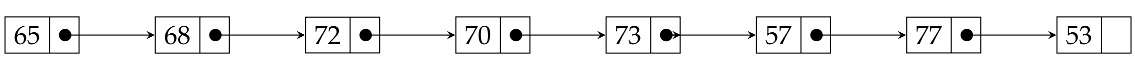


Figure 10: *Included values in the List of nodes to visit after the second algorith step.*

As we start the second visit from the note 65, the first interval will be an ascending perfect fourth (+5) and the second, descending perfect fourth (-5). As we check the descending perfect fourth, we will note that the resulting note (60) has been previously visited and, for that reason, will not be placed in line to be visited again. The algorithm operation can be visually represented as demonstrated in Figure 11.

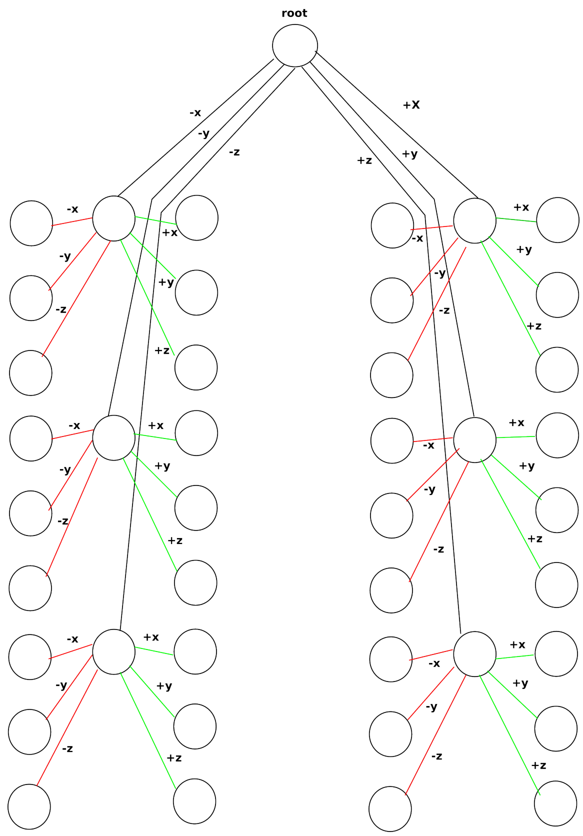


Figure 11: *Visual representation of the algorithm.*

The third and last created data structure is the one that represents the path to each one of the notes. We use the representation of a tree for this structure. The trees are abstract data structure that can be represented in several different ways. Independently from the kind of representation, every tree has a root note (central note) and several offspring that can be reached from the center. The nodes that have not produced offspring are called leaves and the nodes that are not the central node, but that have produced offspring, are called branches. A tree data structure must be navigable starting from its root node, in a way that all the other nodes can be reached from this base root node. The connections between the nodes can have a label and the sum of these labels is to indicate which path from the central node was taken in order to reach any other node.

Because the universe of nodes is finite, an initial limitation for this implementation, we can represent a tree by means of a vector (or array). This way, each one of the notes is marked in order to show which path must be taken to reach the root of the tree. We start by marking the root with a label R (root). The other intervals are marked by the labels *f* and *F* (first), *s* and *S* (second), and *t* and *T* (third), being that the upper case indicates the ascending intervals and the lower case, the descending one. Thus, our path tree, after the first interaction, can be represented by Figure 12.

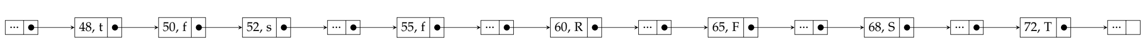


Figure 12: *Path tree after first algorithm step.*

This representation indicates that, in order to depart from note 55 and reach the root node, we have to move an ascending perfect fourth (inverse path to the one stored in the vector). In order to move from note 50 to the root, we move an ascending perfect fourth (f) to note 55, and another ascending perfect fourth (f) to note 60. This way, the path from the note to the root will be (ff). Since each note will only keep one path to reach the root (as determined by the algorithm), the representation in array is enough and has a low computational cost. Note that we are using a relative representation of the network, that is, we represent the first, the second, and the third interval, independently from the choice of intervals for the axes, what turns such piece of information into a metadata that will be used for the exchange of information between distinct networks.

After the algorith have created the whole data, the Harmonic Network, using the structure *list of nodes to visit*, *visited nodes*, and the *path tree* , we will use only this last to the operations that relates different Harmonic Networks. The algorithm uses the *Tree of patches* in a reverse order to found the patch of one note to the central note and stop running, *posting* on screen the simbolic representation of the patch traversed in the Harmonic Network structure.

ii. Using the Harmonic Networks

In order to illustrate the network operation in SuperCollider, let's take a look at the example given by Pousseur in his text, demonstrating how it is obtained by using the implementation presented here. Pousseur makes use of the initial part of the integrationist song *We Shall Overcome*, a symbol of the North American African-American resistance, as reproduced in the first system of Figure 13 and with repetitions eliminated in the second line of the figure.

Figure 13: *Excerpt of the song We Shall Overcome. Adapted from Pousseur (2009).*

Pousseur places the notes of these chords in an initial network (that will be referred to as original), constituted by the X, Y, and Z axes with perfect fifth, perfect octave, and major third intervals respectively. Then, he remaps the same positions in a network with axes X, Y, and Z transformed, respectively, in perfect fifth, major seventh, and major third.

Such operation is accomplished in our implementation as follows (using the notes of first chord):

```
(
var original, transformed, positions, notes;
original = Pousseur(60, 7, 12, 4);
positions = original.notes2path([48, 55, 60, 64, 67]);
transformed = Pousseur(60,7, 11, 4);
notes = transformed.path2notes(positions);
)
```

By executing the code above, the result will be:

```
[ 49, 56, 60, 64, 67 ]
```

Describing the code, line by line:

The initial and final parentheses that limit part of the code and the first line that creates the variables where the operations resulting values are kept are requirements of the SuperCollider language that will not be approached here. In the third line, we create the original network, determining the network central note (60 = middle C in frequencies expressed in MIDI values) and the X, Y, and Z axes as: 7 = perfect fifth, 12 = perfect octave, and 4 = major third, respectively. In the following line, we request that the function *notes2path* find notes 48, 55, 60, 64, and 67 (respectively C3, G3, C4, E4, and G4 in MIDI pitches) in the original network and save the result under the *positions* variable. After that, we create the transformed network with the central note also in C4 and axes: perfect fifth, minor seventh, and major third. Finally, we request that the function *path2notes* search in the transformed network for positions that were stored in the *positions* variable, obtained from the original network, resulting in the transformed notes: 49, 56, 60, 64, and 67, respectively: D♭3, A♭3, C4, E4, and G4.

This code strand demonstrates the basic use of the implementation of networks, but it is not the implementation itself. The SourceCode is available in the following address: https://sourceforge.net/projects/scpousseur/, either for SuperCollider as for PureData.

V. Conclusion

After framing the main question in the introduction of this paper, which is "how can we create live electronic music that incorporates both temporal malleability and performance freedom, without disregarding the structural speculation of musical materials?", we can now present Pousseur's Harmonic Network as a possible answer. Harmonic Networks are a versatile tool for creating musical structures that, as presented in this text, can relate both the Harmonic and Melodic materials in several complex manners, making it possible for sound artists to find a way to generate music materials for live-electronics, regarding Harmonic Relations at the same complexity level as the high levels of elaboration in the "instrumental dimension" (remembering Bruno Maderna). The proposal presented here does not intend to invalidate or criticize the improvisational or the Sonification process so current in live-electronics. It only aims at demonstrating other processes that include the harmonic speculation of structures to be applied in this musical poetics.

In several live-electronic works that I composed in past years, I have used Harmonic Networks in several different varied procedures. Some works use the Harmonic Networks associated with information from amplitude sensors, rhythmic density analyzers, and others, to create several layers of harmonic and melodic synthesis that are related to the instrumental dimension. In some other works, Harmonic Networks are related with algorithms that classify, in real-time, the overall harmonic dissonance and consonance (Harmonic Tension) resulting from Harmonic Networks transformations and associate their results with several kinds of information detected from the instrumental performance. Such algorithm of Harmonic consonance and dissonance classification is though a topic for a next paper.

References

[1] Belet, B. (2003). Live Performance for Humans and Machines in the Early Twenty-first Century: One Composer's Aesthetics for Composition and Performance Practice. *Organized Sound*, 8/3, pp. 305–312.

[2] Dias, H. (2006). *"A Querela dos Tempos": um estudo sobre as diverências estéticas na música eletroacústica mista.*. Dissertation (Master's in Music), UNESP, São Paulo.

[3] Domingues, D. (2009). *Arte, ciência e tecnologia*: Passado, presente e desafios. São Paulo: Editora UNESP.

[4] Emerson, S. (2010). *Living Electronic Music*. Cornwall: Aschgate.

[5] Garnett, G. (2001). The Aesthetics of Interactive Computer Music. *Computer Music Journal*, 25/1, pp.21–44.

[6] Kimura, M. (2003). Creative Process and Performance of Interactive Computer Music. *Organized Sound*, 8/3, pp. 289–296.

[7] Machover, T. (1992). *Hyperinstrument*: A Progress Report 1987–1991. Massachusetts, MIT Media Laboratory.

[8] Magnusson, T. (2011). Algorithms as Scores: Coding Live Music. *Leonardo Music Journal*, 21, pp.19–23.

[9] Manoury, P. (2007). *Considerátions (toujours actuelles) sur l'état de la musique en temps réel*. Available in http://www.philippemanoury.com/?p=319

[10] Maturana, H. et al. (1997). *A ontologia da realidade*. Belo Horizonte: UFMG.

[11] McNutt, E. (2003). Performing Eletroacustic Music: A Wider View of Interactivity. *Organized Sound*, 8/3, pp.297–304.

[12] Menezes, F. (1997). *Atualidade estética da úsica eletroacústica*. São Paulo: Editora UNESP.

[13] Petitot, F. et al. (1996). *Naturalizing Phenomenology*. Stantford: Stanford University Press.

[14] Pousseur, H. (1998). Applications analytiques de la "techiques des réseaux". *Revue Belge de Musicologie*, 52, pp.247–298.

[15] Pousseur, H. (2009). *Apoteose de Rameau*: e outros ensaios. São Paulo: Editora UNESP.

[16] Riemann, H. (1992). Ideas for a Study "On the Imagination of Tone". *Journal of Music Theory*, 36/1, pp.81–117.

[17] Risset, J. (1999). Composing in Real-time? *Contemporary Music Review*, 18/3, pp.31–39.

[18] Risset, J. and Duyne, S.(1996). Real-Time Performance Interaction with a Computer- Controlled Acoustic Piano. *Computer*, 20/1, pp.62–75.

[19] Rowe, R. (1993). *Interactive Music Systems*. Boston: The MIT Press.

[20] Rowe, R. (2001). *Machine Musicianship*. Boston: The MIT Press.

[21] Schoenberg, A. (2004). *Funções estruturais da harmonia*. São Paulo: Via Lettera.

[22] Di Scipio, A. (2003). "Sounds in Interface": From Interactive to Ecosystemic Signal Processing. *Organized Sound*, 8/3, pp.269–277.

[23] Varella, F. et al. (2003). *A mente incorporada*. Porto Alegre: Artmed.

[24] Wilson, S. et al. (2011). *The SuperCollider Book*. Boston: The MIT Press.

A few more Thoughts about Leibniz: The Prediction of Harmonic Distance in Harmonic Space

Michael Winter
mwinter@unboundedpress.org

Abstract: *This paper shows how, over 300 years ago, Gottfried Wilhelm Leibniz envisioned James Tenney's theory of harmonic distance in harmonic space. A description of Tenney's theory is followed by an analysis of musical ideas contained in letters from Leibniz to Christian Henfling. The analysis compares and contrast Leibniz's ideas to Tenney's ultimately showing that they are practically identical.*

Keywords*: Gottfried Wilhelm Leibniz, James Tenney, harmonic distance, harmonic space, complexity.*

I. Introduction

i. The current discourse on Leibniz's writings about music

Gottfried Wilhelm Leibniz was a polymath of epic proportions. He lived in a time where the notion of the specialist had not become ubiquitous as it is today. His objects of study were the world and most everything in it. As such, it should be of no surprise that he also thought and wrote extensively about music.

While many consider the *Monodology* his most seminal work, Leibniz does not have a single, large-scale text that has singularly defined him historically (say, for example, on the order of Baruch Spinoza's *Ethics*). The *Monodology*, which is indeed an incredibly profound treatise despite its brevity, hardly encompasses the scope of Leibniz's intellectual curiosity.

Most of Leibniz's writings exist in the form of letters and those on music are no exception. History has favored his mathematical and philosophical work whereas the music related texts have only been discussed by a handful of researchers. Most notably, the published findings of Walter Bühler [1, 2] and Andrea Luppi [3] are an invaluable resource for entry into Leibniz's musico-intellectual world. Bühler and Luppi have done extraordinary work citing and providing detailed explanations of Leibniz's music-related texts.

Bühler and Luppi's analysis of Leibniz's musical ideas are contextualized within a music-theoretical discourse that has persisted from Leibniz's time to today. Theories of functional tonal harmony, tuning, and aesthetics developed by Leibniz's immediate contemporaries (for example, Johann Joseph Fux, Jean-Philippe Rameau, and Andreas Werckmeister) continue to serve as a foundation for music analysis. Given that Leibniz lived in a time of incredible musical development, exploration, and experimentation, it is easy to underestimate the progressiveness of Leibniz's own musical ideas. Leibniz was by no means just a curious observer. Rather, he was an ardent contributor to the rapidly evolving discourse.

Bühler and Luppi have done great service bringing many of Leibniz's musical ideas to light. But, as mentioned above, their work is situated within a specific theoretical and musicological context which does not extend to more recent trends in music. As with other intellectual domains, I posit that Leibniz was even more forward-thinking with respect to music than some of his immediate contemporaries. He predicted musical ideas that have yet to enter the theoretical canon. Musical ideas explored by composers of *our* time, not *his*.

ii. Genesis of an unlikely discovery: connecting Leibniz to Tenney

2016 marked the 300th anniversary of Leibniz's death. Through the suggestion of Greg Chaitin, with whom I have been close friends with for many years, Ugo Pagallo invited me to participate in the 'Leibniz's Vision' conference which was hosted in Turino, Italy 300 years to the day of Leibniz's death. I still sometimes smirk at the thought that the conference would have been better served had Bühler or Luppi been invited. After all, they are the world's leading experts on Leibniz's writings about music. I, on the other hand, am just a composer that got to Leibniz rather backwards via my connection to Greg Chaitin and an ongoing interest in algorithmic information theory and complexity.

Originally Ugo and I envisioned my contribution to the conference less as a scholarly one and more so as a commission to write a piece celebrating Leibniz. Still, after receiving the invitation, I completely immersed myself in Leibniz's writings looking feverishly for references to art and music. In the early stages of my research, well before I became familiar with the work of Bühler and Luppi, I wrote a piece titled *preliminary thoughts* [7]. In an homage to the fact that Leibniz was a fervent letter writer, *preliminary thoughts* is a 'musical letter' to Greg Chaitin discussing my preliminary reactions to the ideas and writings of Leibniz which I thought directly related to music; specifically, combinatorics, harmony, aesthetics, structure, epistemological vs. practical limits, free will, and even love with respect to creativity. In the piece, a reading of the text of the letter sounds against a minimal guitar part that continually repeats a set of 6 tones with ever changing durations between the articulations of the tones. The guitar part and the reading of the letter are juxtaposed with random flickerings of computer-generated tones and noises.

My idea at the time was that, after doing further research, I would integrate more 'conclusive thoughts' into a new piece that I would eventually perform in Turino. However, our initial thoughts are often the most poignant. While the text of my *preliminary thoughts* is rather informal (like much of the writings of Leibniz), I think it is actually quite comprehensive with respect to addressing many ways in which Leibniz predicted musical ideas well beyond those that have entered into the current canon of music theory.

As the anniversary neared, I struggled to come up with ideas for another, more conclusive piece to present at the conference. In the meantime, I had written an orchestra piece, *essay on the art of combinations* [8], integrating ideas on combinatorics, expressed by Leibniz through music, in his dissertation *On the Art of Combinations*. The score of the piece includes a short essay on how I incorporated Leibniz's ideas. Even though writing the piece was an integral part of what I now affectionately refer to as 'my year with Leibniz', it was still a project ancillary to the conference.

In the weeks preceding the conference, while on tour in Europe and still unsure of what I would do for the conference, I had the opportunity to visit the Leibniz archive. Thanks to the library staff, particularly Michael Kempe and Werner Ganske, as well as the research by Bühler and Luppi, I was able to go directly to the source.

In particular, the correspondence between Leibniz and and Christian Henfling caught my attention as many of the figures and diagrams within looked uncannily similar to figures and diagrams in the writings of one of my mentors, James Tenney. Tenney, like Leibniz, was incredibly

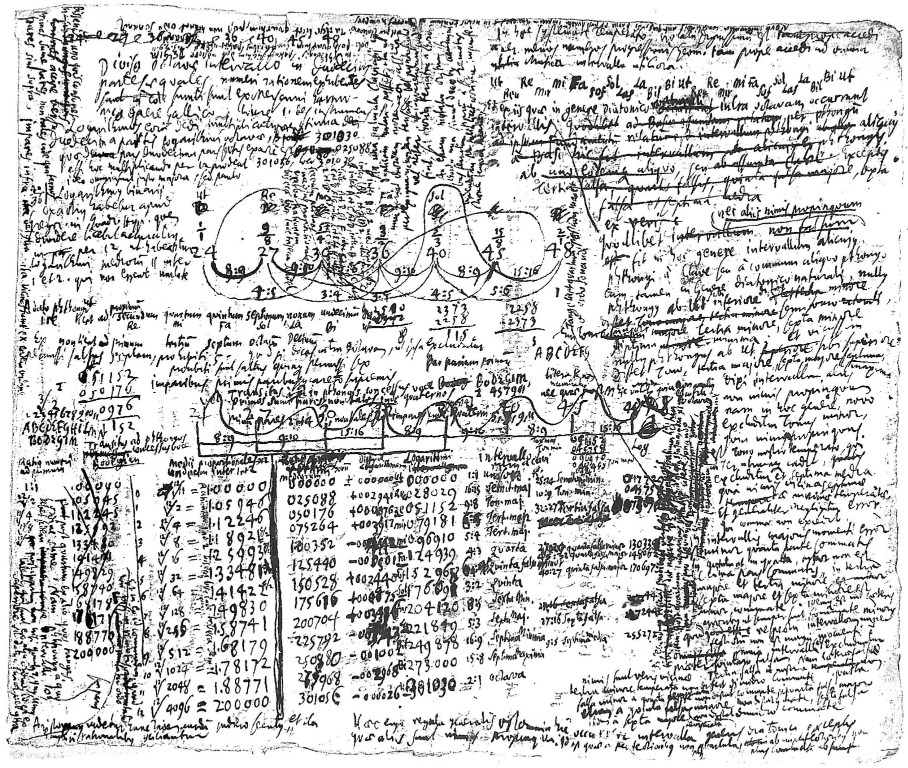


Figure 1: *Example page of one of Leibniz's letters to Henfling.*

prolific and in my humble opinion, one of the most important composers and theorists of the last 100 years.

One of Tenney's primary interests was creating and defining a phenomenologically-based theory of music analysis. His seminal treatise, *Meta ≠ Hodos* [4], written as a graduate student, applies concepts of gestalt psychology to musical analysis. Later, Tenney would combine his interest in phenomenology with an ever-growing interest in harmony. In a series of papers, "The structure of harmonic series aggregates" [5] and "John Cage and the theory of harmony" [6], Tenney defines the concept of "harmonic distance", which is a metric in what he calls "harmonic space". Harmonic distance is essentially an integer complexity function used to create a notion of distance between the frequencies of two tones, often referred to loosely as the level of 'consonance' or 'dissonance'.

Leibniz's letters are difficult to read and decipher (especially for someone whose Latin and French are non-existent). Some of the pages of Leibniz's letters to Henfling are filled with calculations, tables, and diagrams to the extent that there almost seems to be more ink than whitespace (for example, see Figure 1). It might be easy to quickly dismiss Leibniz's music-theoretical attempts as turning music into numbers without taking perception into account. However, after closer examination, I began to realize that over 300 years ago, Leibniz predicted Tenney's theory of harmonic distance in harmonic space.

After making the connection between Leibniz and Tenney, I finally (less than two weeks before the anniversary of Leibniz's death) had more conclusive thoughts and knew for sure what I would do for the conference. I would divide my time between a lecture and a performance: the first part giving my recent findings and the second giving my *preliminary thoughts*.

Ultimately, I think my invitation to the conference turned out to be rather fitting. After all, the central theme was about Leibniz's 'vision' and how it still resonates today. And while I am not a Leibniz scholar, my vantage-point as a working artist/composer helped reveal that Leibniz actually predicted ideas that, as I mentioned previously, are only now being developed more thoroughly. Ideas that have yet to broadly reach the academic world. In fact, it almost seems rather serendipitous. The connections I make in *preliminary thoughts* and Leibniz's prediction of Tenney's theory would have easily gone overlooked if Ugo had not taken a chance in inviting someone like myself, who, in the influence of Leibniz, happily traverses intellectual domains with abandon. Admittedly, I was among such luminaries at the conference that I felt both humbled, honored, and definitely out of place! I am grateful to both Greg and Ugo for including me.

What follows is a more thorough description of harmonic distance in harmonic space followed by an analysis of Leibniz's letters to Henfling demonstrating how he predicts Tenney's theory. The text of my piece *preliminary thoughts* is provided as an appendix.

II. Leibniz's prediction of harmonic distance in harmonic space

i. James Tenney's definitions

The fundamental tenet of Tenney's theory of harmonic distance is that harmonic relations between pitches can be modeled by a multidimensional space with metrical and topological properties that reflect how the human auditory apparatus perceives relations between pitches. In the model, pitches are represented by points in a multidimensional lattice where the dimensions correspond to the prime factors required to specify the frequency ratios of the set of pitches with respect to a given reference pitch. Tenney's own examples are provided in Figure 2 and Figure 3. Note that, as in Figure 3, Tenney often "collapses" (or omits) the 2-dimension as it represents intervals of an octave. Omitting the 2-dimension eliminates duplication of pitch-classes and allows higher dimensions to be more easily plotted.

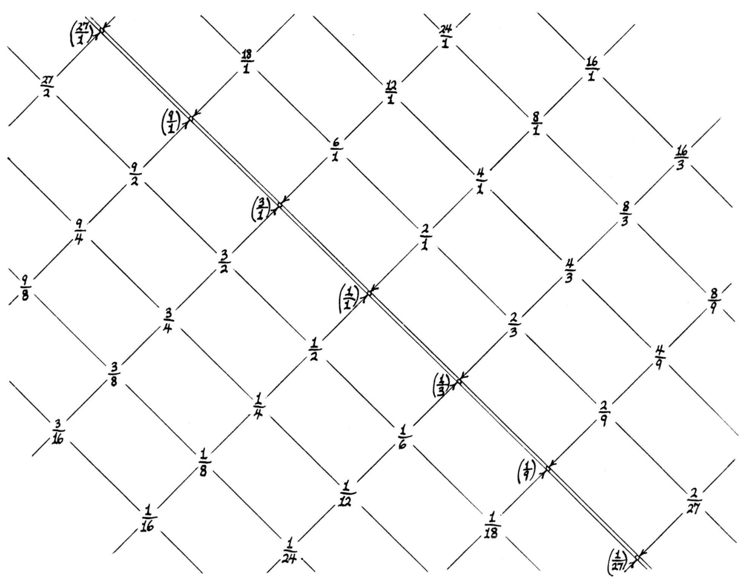


Figure 2: *Harmonic lattice in 2,3 harmonic space from Tenney's "John Cage and the Theory of Harmony". [6]*

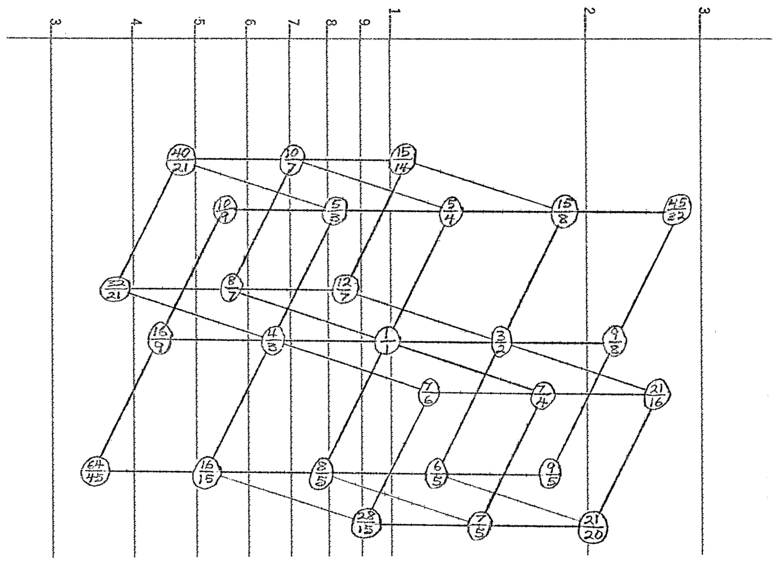


Figure 3: *Harmonic lattice in 3,5,7 harmonic space with the 2-dimension collapsed from Tenney's "The structure of harmonic series aggregates". [5]*

The perceived harmonic distance between two pitches is the distance of the shortest path between the corresponding points in harmonic space. Because harmonic space is a lattice, harmonic distance is a non-euclidean 'city-block' metric.

Tenney's mathematical formulation is beautifully elegant: $HD(a, b) = \log_2(ab)$
where a/b is a frequency ratio such that a and b are coprime.

It is clear to see that harmonic distance is a wonderfully concise quantification of a walk in harmonic space that weights the size of the prime factors because for $ab = 2^i 3^j 5^k \ldots$,
$\log_2(ab) = i \log_2(2) + j \log_2(3) + k \log_2(5) + \ldots$

Note that the collapse of the 2-dimension, as in Figure 3, is somewhat deceiving because information encapsulated in Tenney's harmonic distance function—specifically movement in the 2-dimension—is lost. The collapsed visualization implies that a step in harmonic space is the logarithm of a prime over the highest power of two less than the given prime: $p/2^{\lfloor \log_2(p) \rfloor}$. However, the steps in actual harmonic space are always strictly the logarithm of a prime and the harmonic distance function always computes the steps in *all* dimensions. This is why, for example, 3:2 and 4:3 do not have the same harmonic distance. This can be seen by comparing the number of steps from 1:1 to 4:3 in Figure 2, which includes the 2-dimension, in comparison to Figure 3, which does not.

As mentioned in the previous section, Tenney's harmonic distance function is an integer complexity function based on the number, size, and exponents of the prime factors needed to represent the frequency ratio between two pitches. I often explain Tenney's harmonic distance function in the context of computational complexity: that Tenney's formulation of harmonic distance quantifies the amount of time it takes to compute the prime factors of a number. That is, assuming a Leibnizian digital philosophy where the brain is a sophisticated computer, two pitches are perceived as being more closely related (and therefore also closer in harmonic space) because it takes less time for the brain to compute the frequency ratio between them.

ii. Leibniz's vision

What first caught my attention in Leibniz's correspondence to Henfling is an identity function expressing when two pitches are the same (see Figure 4). The equivalence is written in terms of the prime factors and their respective exponents. Written below the identity function is a set of numbers plotted in a 2-dimensional table where the vertical and horizontal dimensions are powers of 2 and 3, respectively. At this point, the connection to Tenney's concept of harmonic space became clear as Leibniz's table is equivalent to a harmonic lattice in 2,3 harmonic space.

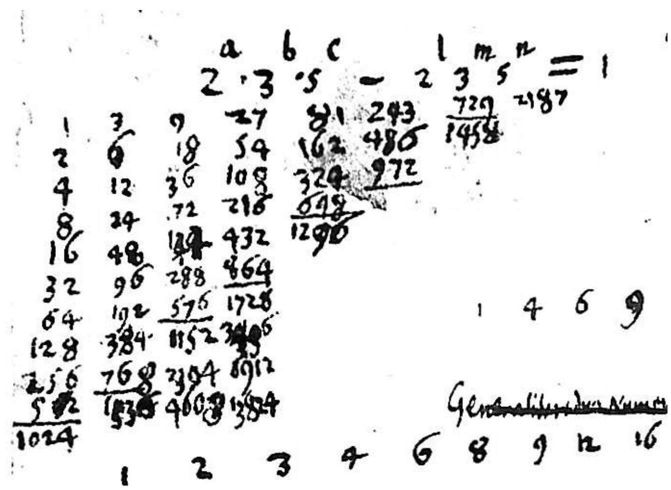


Figure 4: *Leibniz's identity function and 2,3 harmonic space.*

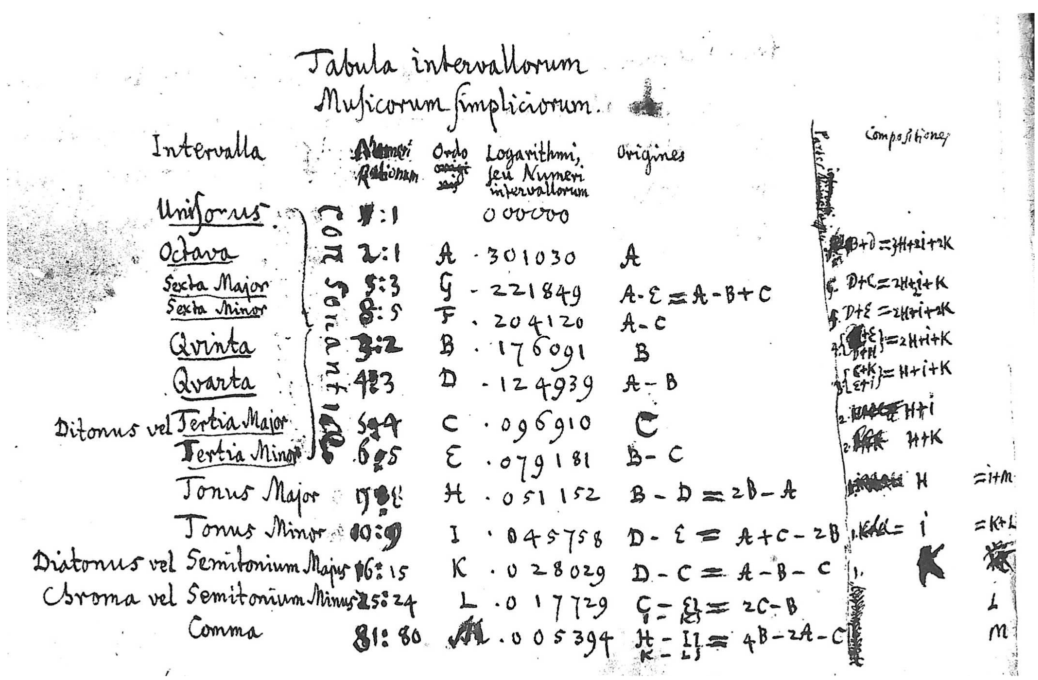


Figure 5: *Leibniz's Tabula intervallorum Musicorum Simpliciorum.*

But the connection did not stop there. A following page in the correspondence includes a table Leibniz titles "Tabula intervallorum Musicorum Simpliciorum" (see Figure 5). Each row of the table corresponds to a musical interval listed in order from big to small with respect to the perceptual size of the interval (the logarithm of the interval). If Leibniz's prediction of harmonic space suggested by Figure 5 was not enough, his Tabula intervallorum seems to be a near verbatim expression of Tenney's harmonic distance function.

Of particular interest are the 3rd and 5th columns: "Ordo" and "Origines". The 1st, 2nd, and 4th are "Intervalla" - interval name, "Numeri Rationum" - frequency ratio, and "Logarithmi seu Numeri intervallorum" - logarithm to the base 10 of the frequency ratio, respectively. In the Ordo column, Leibniz assigns each interval an alphabetical index. While the ordering function is not explicated, I believe Leibniz means 'order of consonance'. He also groups the first 8 intervals and labels them as "Consonantie" (or the consonances). The Origines column expresses each interval as the sum of the 3 most consonant intervals: A - 2:1, B - 3:2, and C - 5:4 - the octave, perfect fifth, and major third, respectively.

Again, what is not completely clear is how Leibniz derives his Ordos. Perhaps it was empirical. However, we arrive at Tenney's harmonic distance function by substituting the terms xA, yB, and zC by $x\log_2(2)$, $-y\log_2(2)+y\log_2(3)$, $-2z\log_2(2)+z\log_2(5)$, respectively; combining like terms; then replacing the resulting coefficients with their absolute values (as shown below). With two exceptions (marked with an asterisks in Table 1), the fact that Leibniz's Ordo corresponds to the order of the intervals sorted by Tenney's harmonic distance function follows.

$$
\begin{aligned}
&xA + yB + zC \\
\text{substitute}\quad & \\
&x\log_2(2) + (-y\log_2(2) + y\log_2(3)) + (-2z\log_2(2) + z\log_2(5)) \\
\text{combine}\quad & \\
&(x-y-2z)\log_2(2) + y\log_2(3) + z\log_2(5) \\
\text{replace}\quad & \\
&|x-y-2z|\log_2(2) + |y|\log_2(3) + |z|\log_2(5)
\end{aligned}
\tag{1}
$$

Table 1: *Leibniz's Origines expressed as Tenney's harmonic distance.*

Freq. ratio	"Origines"	Harmonic Distance	
1 : 1	identity/unity		
2 : 1	A	$\log_2(2)$	$= 1$
3 : 2	B	$\log_2(2)+\log_2(3)$	$= 2.58496$
5 : 4	C	$2\log_2(2)+\log_2(5)$	$= 4.32193$
4 : 3*	$A-B$	$2\log_2(2)+\log_2(3)$	$= 3.58496$
6 : 5	$B-C$	$\log_2(2)+\log_2(3)+\log_2(5)$	$= 4.90689$
8 : 5	$A-C$	$3\log_2(2)+\log_2(5)$	$= 5.32193$
5 : 3*	$A-B+C$	$\log_2(3)+\log_2(5)$	$= 3.90689$
9 : 8	$2B-A$	$3\log_2(2)+2\log_2(3)$	$= 6.16993$
10 : 9	$A+C-2B$	$\log_2(2)+2\log_2(3)+\log_2(5)$	$= 6.49185$
16 : 15	$A-B-C$	$4\log_2(2)+\log_2(3)+\log_2(5)$	$= 7.90689$
25 : 24	$2C-B$	$3\log_2(2)+\log_2(3)+2\log_2(5)$	$= 9.22882$
81 : 80	$4B-2A-C$	$4\log_2(2)+4\log_2(3)+\log_2(5)$	$= 12.6618$

Computing the necessary replacements in order to arrive from Leibniz's formulation to Tenney's harmonic distance function is rather straightforward. B - 3:2 is the equivalent of one step forward in the 3-dimension and one step back in the 2-dimension (by back, I mean its reciprocal, $1/2$): $3/2 = 3/1 \times 1/2$. 5:4 is the equivalent of one step forward in the 5-dimension and two steps back in the 2-dimension: $5/4 = 5/1 \times 1/2 \times 1/2$. Similar to Tenney's collapsed representation, Leibniz is incorporating movement from the 2-dimension in the other dimensions/terms in order to bring all the frequency ratios within an octave (a ratio between 1 and 2). The replacements above expand out the movement in the 2-dimension encoded in the other dimensions. The fact that Leibniz does not completely remove the 2-dimension (by maintaining the term A - 2:1) is the very reason that it is possible to show that his formulation was similar to, if not completely the same as, Tenney's harmonic distance function. By doing so, Leibniz uses steps that are within an octave (that is, logarithms of primes over the highest power of two less than the given prime) while maintaining a way to compute the movement in the 2-dimension needed to specify the ratio. The very movement, as explained in the previous section, that is lost in Tenney's collapsed visualizations yet preserved in his harmonic distance function. Figures 6, 7, and 8 demonstrate Leibniz's Origins in what I call a Leibnizian harmonic lattice (where a step is $p/2^{\lfloor \log_2(p) \rfloor}$ with exception of the term A - 2:1), in one of Tenney's 3,5 lattices with the 2-dimension collapsed, and finally in actual 2,3 harmonic space.

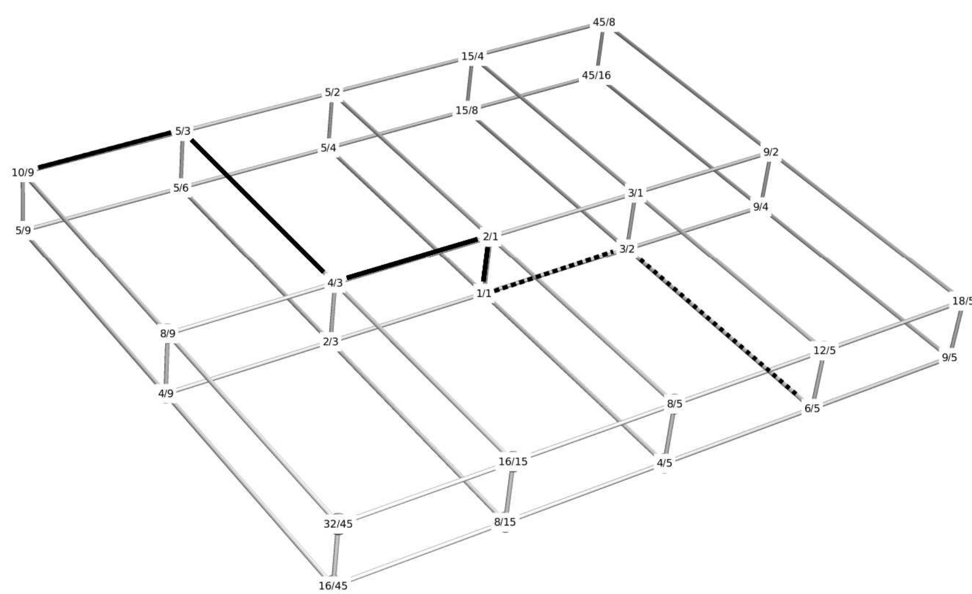


Figure 6: *Leibniz Origines for 10:9 (solid line) and 6:5 (dotted line) as paths on a Leibnizian harmonic lattice.*

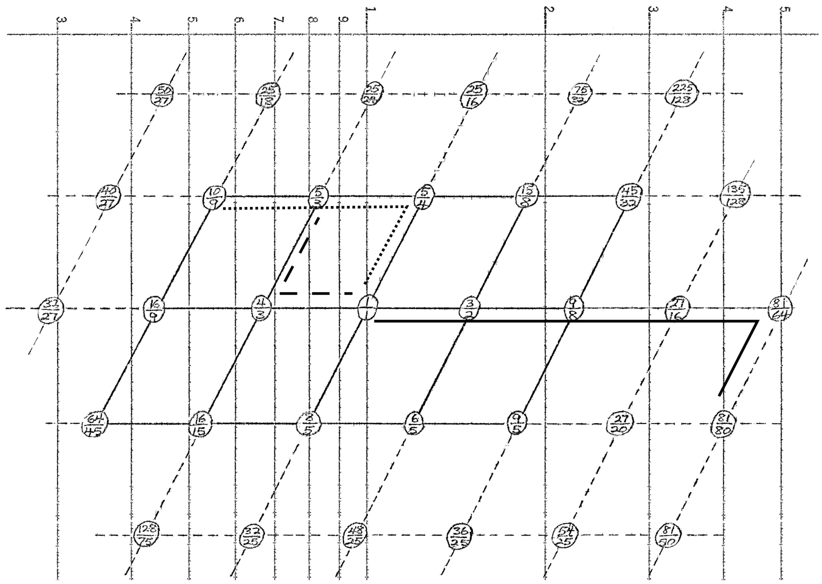


Figure 7: *Leibniz Origines for 81:80 (solid line), 10:9 (dotted line), and 5:3 (dashed line) as paths on Tenney's harmonic lattice in collapsed 3,5 space from "The structure of harmonic series aggregates". (Note that the A terms from Leibniz's Origines have been omitted.)*

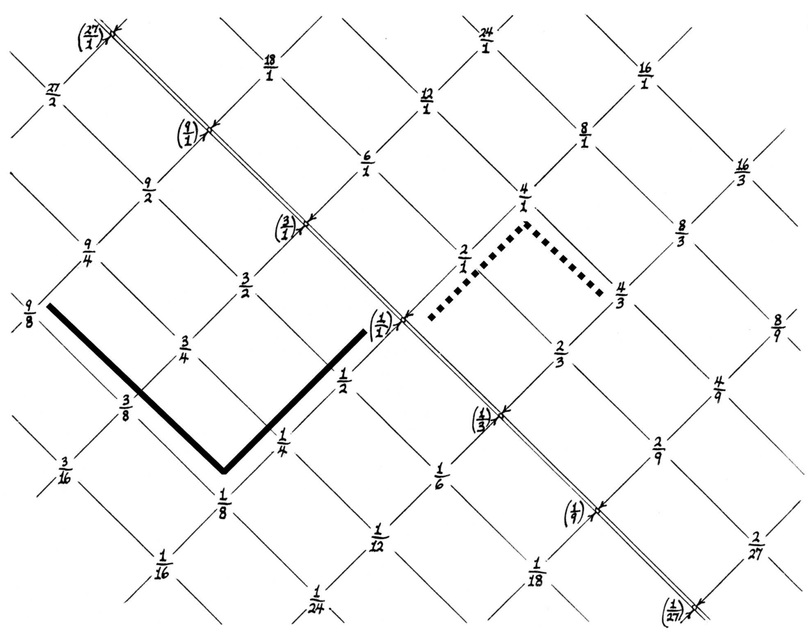


Figure 8: *Leibniz Origines for 9:8 (solid line) and 4:3 (dotted line) expanded out as paths on Tenney's harmonic lattice in actual 2,3 harmonic space from "John Cage and the theory of Harmony". (Note that the numbers of steps in each dimension is equal to the coefficients in the long form expression of the harmonic distance function in Table 1.)*

III. Conclusion

This is a very specific case of Leibniz's vision over 300 years ago that applies to music and theories being advanced today. The text of my piece *preliminary thoughts* gives other examples, but in less depth. With Leibniz, you always get the sense that you are only scratching the surface. Apart from the correspondence with Henflig, Leibniz wrote about music in *The Art of Combinations* and to several others such as Christian Goldbach, Christiaan Huygens, Joseph Sauveur, and Agostino Steffani. Again, I refer the reader to the work of Bühler and Luppi for a comprehensive overview.

It is unlikely that all of Leibniz's writings about music have been accounted for. The Leibniz archive contains copies of letters that he wrote and kept, but copies that he did not keep likely ended up solely in the possession of the correspondee. If someone were so diligent as to look into the archives of those with whom he corresponded, I imagine that there would be more intellectual treasures to be found, especially with respect to music.

References

[1] Bühler, W. (2010). Musikalische skalen und intervalle bei leibniz unter einbeziehung bisher nicht veröffentlichter texte (teil I). *Studia Leibnitiana*, 42(2), pp.129-161.

[2] Bühler, W. (2012). Musikalische skalen und intervalle bei leibniz unter einbeziehung bisher nicht veröffentlichter texte (teil II). *Studia Leibnitiana*, 44(2), pp.134-165.

[3] Luppi, A. (1989). *Lo Specchio Dell'armonia Universale Estetica E Musica in Leibniz.* Milano: Franco Angeli.

[4] Tenney, J. (2015). Meta ≠ Hodos. In: *From Scratch: Writings in Music Theory.* L. Polansky, L. Pratt, R. Wannamaker, M. Winter (eds.) Champaign, IL: University of Illinois Press, pp. 13-96

[5] Tenney, J. (2015). The structure of harmonic series aggregates. In: *From Scratch: Writings in Music Theory.* L. Polansky, L. Pratt, R. Wannamaker, M. Winter (eds.) Champaign, IL: University of Illinois Press, pp. 240-279

[6] Tenney, J. (2015). John Cage and the theory of harmony. In: *From Scratch: Writings in Music Theory.* L. Polansky, L. Pratt, R. Wannamaker, M. Winter (eds.) Champaign, IL: University of Illinois Press, pp. 280–304

[7] Winter, M. (2016). *preliminary thoughts.* Unbounded Press. Musical score available on the web at http://www.unboundedpress.org/scores/preliminary_thoughts_score.pdf.

[8] Winter, M. (2016). *essay on the art of combinations.* Unbounded Press. Musical score available on the web at http://www.unboundedpress.org/scores/essay_on_the_art_of_combinations_score.pdf.

preliminary thoughts

> The following text is a 'musical letter'. The full score which includes performance instructions, the musical notation for for the guitar part, the code for the computer generated accompaniment, and a Spanish translation of the text by Nicolás Carrasco Diaz is available at:
> http://www.unboundedpress.org/scores/preliminary_thoughts_score.pdf

Dear Greg,

As I mentioned in prior correspondence, in consideration of the upcoming celebration of Leibniz on the 300th anniversary of his death, I have immersed myself in his work; reading and rereading his texts as much as time allows. His oeuvre is so voluminous, that I fear even by the time we meet in November, I will have only scratched the surface.

I have been enjoying the fact that much of Leibniz's writings are in the form of letters. They are less precious, less formal in that way. As I prepare to write the piece for the celebration in Turin, I thought it would be nice to set my correspondence with you to music. As musical letters or studies of sorts. Ideas not yet fully formalized but worth expressing; both the text and the accompanying music.

I write this letter as an exposition of my preliminary reactions in hopes that the very articulation and expression of these thoughts will aid in their future formulation albeit as naive as they may be in their current state.

In Leibniz's writings, I have found several cogent threads that intrinsically (if not explicitly) relate to art and music. I will group them as follows even though they are all interrelated: combinatorics, harmony, aesthetics, structure, epistemological vs. practical limits, and free will.

1) Combinatorics

I found Leibniz's dissertation entitled *On the Art of Combinations* of particular interest. Perhaps because it is an early work; laden with mistakes yet sound in its conception. But more likely because of explicit references to the application of combinatorics to music. Although it was written for his studies in jurisprudence, it is humbling that it can apply to so many other domains.

My composer friend Tom Johnson first showed me the 6th of 12 problems from the dissertation last summer though I was unaware of the source at the time. In the problem, Leibniz tries to count the number of 6 note melodies that can be sung with 7 possible pitches. He classifies them by the number of repeated elements. That is, he was trying to give a solution for the number of tuples and permutations with prescribed repetitions.

Earlier in the dissertation, he also discusses the application of combinations from problems I and II to organ registry and counts the number of possible timbres that an organ with a certain number of stops can sound (i.e., all subsets of the stops). In this sense, Leibniz predicted over 300 years ago musical ideas that are only now being explored by composers more thoroughly. Though there are important precedents. Bell-ringing traditions come to mind and also the music of Bach, of course. I like to think that there was a sort of intellectual resonance between Leibniz and Bach based on the fact that they lived near each other at the same time. I am also curious if Bach might have been alluding to the title of Leibniz's dissertation in the *Art of the Fugue*.

2) Harmony

While I have yet to find a full version of Leibniz's letter to Christian Goldbach, I have found the following translated excerpts:

> "All our usual intervals are ratios based on two of the prime numbers, 1, 2, 3 and 5. If we were endowed with a little more subtlety, we might arrive at the prime number 7. And actually I believe the following ones are also given. Thus the ancients did not openly avoid the number 7. But hardly anybody proceeded as far as the following prime numbers, 11 and 13."

Then later in the letter he writes:

> "I do not believe that irrational ratios are pleasing to the soul in themselves, except when they are very close to the rational ones which give pleasure."

Clearly Leibniz had a keen understanding of musical harmony. These are deep insights rooted in the Greeks yet only revived recently by composers such as Harry Partch and James Tenney. And indeed, as Leibniz predicted, composers are starting to more thoroughly explore harmonies based on higher prime numbers; what Tenney calls extended harmonic spaces with higher dimensions.

The second quotation might refer to the interleaving of dissonances with consonances as is common in chordal progressions within the rubric of functional tonal harmony. However I prefer another interpretation: that Leibniz is suggesting what Tenney calls "tuning tolerance"—the idea that the brain resolves irrational harmonies to the nearest simplest set of frequency ratios.

Admittedly, I have yet to follow this thread in Leibniz's writings to further extent but hope that I can find more texts that refer to harmony and harmonic constructs.

3) Aesthetics

It is hard to fully understand Leibniz's thoughts on the perception of beauty. He often alludes to the concepts of good and bad with respect to music and art, which I disagree with. In my mind, absolute beauty does not exist. People who believe in it are actually referring to status quo bias where the status quo is the current popular opinion. That is, if someone deems something as universally bad, it actually means that it is against the status quo with which they are in agreement. Whether or not, and how, someone appreciates beauty must be subjective even though biases will arise, especially within cultures. I have theorized in the past what can bring about a person's opinions with respect to if and how they appreciate something they perceive and why this can differ from person to person. I can even demonstrate it in terms of Algorithmic Information Theory, but I will leave that for a later time and remain focused for now on where Leibniz and I align.

In both his "Discourse on Metaphysics" as well as "Meditations on Knowledge, Truth, and Ideas", Leibniz discusses the concepts of "clear" and "confused" knowledge. The latter is of particular interest to me. To paraphrase Leibniz with my understanding of the concept: confused knowledge is the ability to perceive something as distinguished from other things yet unable to express the properties which give rise to its distinction. I sometimes tell people that music often interests me when I know that there is some underlying process even though I cannot identify or properly articulate exactly what that process is. I refer to this as the 'incalculability of concept-to-percept-transparency', which is the inability in art to know to which extent someone can deduce the concept of a work from the perception/experience of it.

4) Structure

Leibniz's discussion on the relation of parts to other parts and to the whole (an example of which I will give later with respect to epistemological vs. practical limits) is almost found verbatim in the composer John Cage's definition of structure. However, Leibniz had even more radical thoughts pertaining to structure. As you have pointed out in your writing, Leibniz basically predicts Algorithmic Information Theory with the following quotation from his "Discourse on Metaphysics":

> "If someone traced a continuous line which is sometimes straight, sometimes circular, and sometimes of another nature, it is possible to find a notion, or rule, or equation common to all the points of this line... When a rule is extremely complex, what is in conformity with it passes for irregular... But God has chosen the most perfect world, that is, the one which is at the same time the simplest in hypothesis and the richest in phenomenon."

This statement is essentially synonymous with the fundamental tenet of Algorithmic Information Theory: that you have structure if the computer program that generates a given object is smaller in bits than the object itself. It is this idea perhaps more than the others that I would like to follow as thoroughly as possible in Leibniz's work to better understand its genesis.

5) Epistemological vs. Practical Limits

In the dissertation, Leibniz writes:

> "The concept of parts is this: given a plurality of beings all of which are understood to have something in common; then, since it is inconvenient or impossible to enumerate all of them every time, one name is thought of which takes the place of all the parts in our reasoning, to make the expression shorter. This is called the whole. But in any number of given things whatever, even infinite, we can understand what is true of all, since we can enumerate them all individually, at least in an infinite time. It is therefore permissible to use one name in our reasoning in place of all, and this will itself be a whole."

Similar to how Leibniz was interested in an alphabet of human thought and the lexicon of a universal language, making art is often about defining elements and how they are (or can be in the case of a more open work) arranged. And just as it is inconvenient to enumerate through all subject-predicate pairs for a universal language, so too is it often difficult, if not altogether impossible, to enumerate all possible musics made from a given set of musical elements. I often find that the musical concepts that I envision in the compositional process quickly spiral out of control in the same way that their more abstract mathematical analogs in combinatorics explode exponentially. But where does the inspiration come that guides the artist to limit the material and order it in a particular way? Here Leibniz's faith in God guides him. Much of his work references the perfection of God's creation and the dissertation itself starts with a proof of God's existence. But this is all in search of truth and clearly he is seeding the idea of a universal proof checker. That is yet another thing that amazes me about his thought process. Almost as asides, he invents new fields of mathematics or prophesizes concepts that are only proved or disproved much later.

This rift between the limits of knowledge and the limits of practicality also occurs in Algorithmic Information Theory. Beyond the paradox of not being able to find a minimal program with certainty, just finding a program that outputs a given result at all is exhaustive beyond our computing means today. I dream of a world in which all my ideas would be computable.

6) Free will

The rift I discuss above also gives me a great deal of faith in intuition and inspiration. And that my intuitive decisions are the very computations I am interested in making with machines. But what is choice? Leibniz believes that all true predicates are contained within a given subject. This is yet another idea where Leibniz and I have independently aligned if I interpret his thoughts correctly. I believe he suggests that because you are unaware of the future, despite its containment in the subjects of the world, that whether or not there is free will does not matter. I have referred to this as the 'illusion of choice' in my own writing. And suggest the very same thing I interpret in Leibniz: that in any world, determinate or not, there is no difference between choice and the illusion of choice.

Then finally, there is love, which I believe must be intrinsically linked to art and creativity. I now know how real love is and how inspired I am by my love for others. Just as art is a "confused" knowledge, so too is love. My body and my senses inform me of its presence and of its loss from another, but my mind cannot explain the reasons for these visceral distinctions. I imagine Leibniz has somewhere discussed what I now understand... that all I do is for love... and that every ounce of my creative energy is for that love to be reciprocated.

With Best Regards,

Michael Winter (Los Angeles; January 23rd, 2016)

FEDERAL UNIVERSITY OF RIO DE JANEIRO

Roberto Leher
Rector

Denise Fernandes Lopes Nascimento
Vice Rector

Leila Rodrigues da Silva
Pro-Rector of Graduate and Research

CENTER OF LITERATURE AND ARTS

Flora de Paoli
Dean

SCHOOL OF MUSIC

Maria José Chevitarese
Director

Andrea Adour
Vice-director

David Alves
Associate Director of Undergraduate Education

Andrea Adour
Coordinator of Bachelor's Degree on Education

Marcelo Jardim
Director of the Cultural Artistic Sector

Ronal Silveira
Director of Extension Course

João Vidal
Coordinator of the Graduate Program in Music

MusMat Research Group

Members:

- Adriel Viturino
- Alexandre Avellar
- Alexandre Ferreira
- Alexandre Schubert
- André Codeço
- André Pinto
- Bernardo Ramos
- Carlos Almada (leader / coordinator)
- Claudia Usai
- Daniel Moreira (coordinator)
- Daniel Mussatto
- Desirée Mayr
- Eduardo Cabral
- Erico Bonfim
- Fabio Adour
- Fabio Monteiro
- Filipe de Matos Rocha
- Gabriel Mesquita
- Helder Oliveira
- Igor Chagas
- João Penchel
- Jorge Santos
- Leandro Chrispim
- Liduino Pitombeira (coordinator)
- Max Kühn
- Marcel Castro-Lima
- Pauxy Gentil-Nunes (coordinator)
- Pedro Miguel de Moraes
- Pedro Proença
- Pedro Zizels Ramos
- Rafael Fortes
- Rafael Soares Bezerra
- Roberto Macedo
- Rodrigo Pascale
- Rodrigo Furman
- Sérgio Ribeiro
- Vinicius Braga