Modeling and Implementing Tiling Rhythmic Canons in the OpenMusic Visual Programming Language

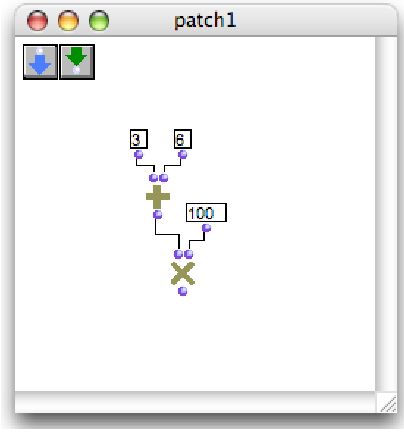
Carlos Agon and
Moreno Andreatta

This article presents tools for the construction of tiling rhythmic canons, implemented in the OpenMusic visual programming language. After a brief presentation of OpenMusic, we will focus on the description of the MathTools environment and its most relevant objects, which enable a composer to produce several classes of tiling rhythmic canons by means of constraint programming, group factorizations, and polynomial representations. Examples of tiling rhythmic canons include canons by translation (from the simplest cases to the cyclotomic canons and Vuza canons) and by augmentation (i.e., canons obtained by affine transformations).

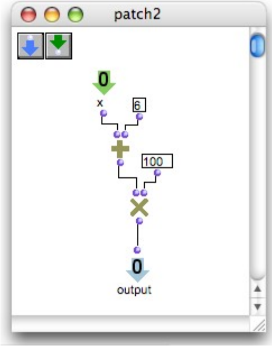
1. The OpenMusic Visual Programming Language and the MathTools Environment

OpenMusic (OM) is a visual programming environment dedicated to music composition, designed and developed at Ircam. It allows for the creation of and experimentation with compositional models through programming by means of a graphical interface that represents musical structures and processes. In order to facilitate the understanding of the illustrations in this paper, the basic concept of *patch* will be explained in this section.[1] A patch is a graphical representation of a program. In a patch, boxes represent functions, each one having a set of inputs (at the top of the box) and a set of outputs (at the bottom of the box). These boxes can be connected together in order to create the functional layout of the program—i.e., what it will do. OM possesses a number of functions of varying complexity and specialization and new ones may be created by the user.

Example 1 shows a simple patch using the functions for the addition and multiplication of integers. It is the graphical equivalent of the expression $(3+6)\times 100$. If the number 3 in the expression $(3+6)\times 100$ is made into a variable, the result is a single-parameter function defined as follows: $f(x)=(x+6)\times 100$. Example 2 shows the patch corresponding to the previous function, where the initial value 3 is replaced by a variable x. From a graphical point of view, this variable is



EXAMPLE 1: A PATCH THAT CORRESPONDS TO THE EXPRESSION $(3+6)\times 100$



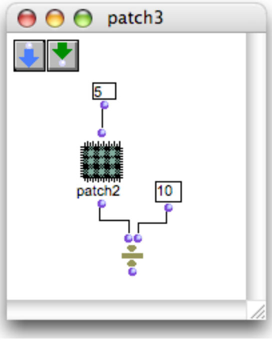
EXAMPLE 2: A PATCH THAT CORRESPONDS TO THE FUNCTION $f(x)=(x+6)\times 100$

represented by an arrow-shaped box (at the top of Example 2) whereas the output arrow (at the bottom) specifies the value that results when the patch is calculated.

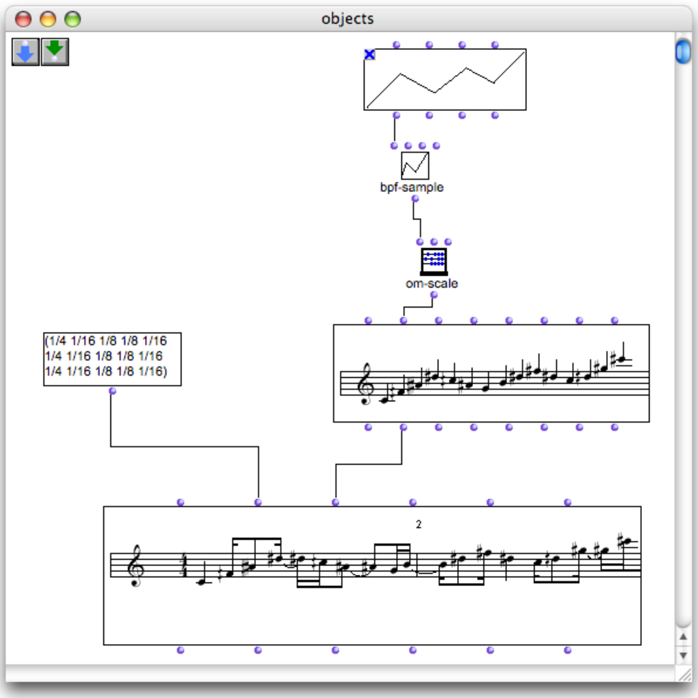
The patch in Example 2 can, in turn, be considered as a function call, and may be used in the form of a graphical box in other patches. This makes it possible to create functional abstractions that can then be used in various contexts. Example 3 shows the patch from the previous example in the form of a box, with the number 5 as its argument. The result of the calculation in this patch will therefore be $f(5)10=((5+6)\times 100)/10=110$. This demonstrates the embedding character of the patch object—i.e., the possibility of nesting a patch within a second patch (and so on).

Data structures (classes) can be used in the patches in order to create and manipulate objects (particularly musical ones). These structures are also shown as graphical boxes with inputs and outputs that provide the program access to their content. Example 4 shows a patch for generating a musical sequence algorithmically. It contains a BPF-type object (break-point function), a chord sequence (represented in absolute time values), and a voice (chord sequence in musical rhythmic notation). Among the OpenMusic objects which are most frequently used by composers are the following musical classes: CHORD, CHORD-SEQ, POLY (polyphony made up of several voices), MIDIFILE, SOUND (audio file), etc.

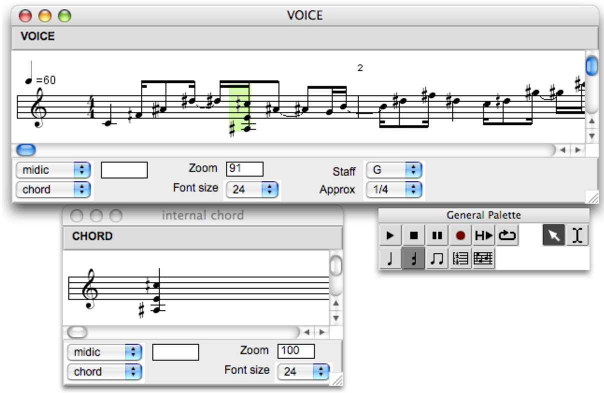
Each object class possesses a graphical editor, enabling the display and manual editing of musical data. Example 5 shows the graphical editor of the voice type object that was constructed in Example 4.



EXAMPLE 3: THE RECURSIVE CHARACTER OF THE PATCH OBJECT



EXAMPLE 4: CREATING A MUSICAL SEQUENCE IN A PATCH WHERE THE PITCHES ARE GENERATED FROM THE DATA IN A BREAK-POINT FUNCTION AND THE RHYTHM IS CREATED OUT OF PROPORTIONAL DURATION RELATIONSHIPS

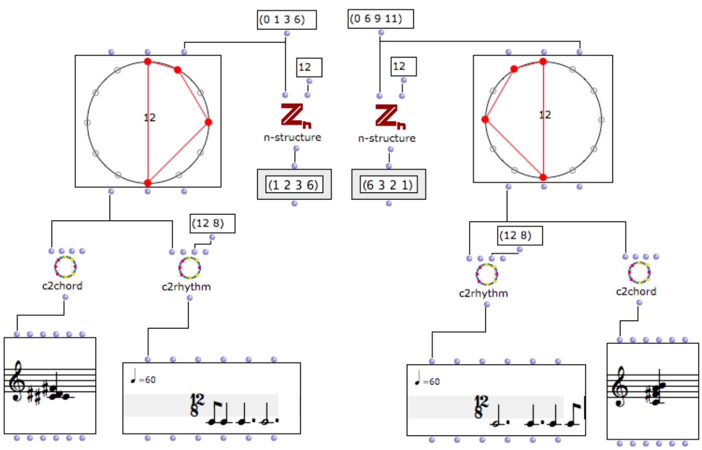


EXAMPLE 5: A VOICE EDITOR

In addition to standard objects in OpenMusic, we developed a collection of mathematical tools based on algebraic structures, allowing music theorists and composers to manipulate new musical material, based on the most recent "mathemusical" research.[2] These tools are organized in a specialized package called "MathTools," which associates the previous musical representations with the circular representation, by means of the N-CERCLE object.

As first observed by a large number of twentieth-century music theorists and composers from the American tradition (Milton Babbitt, David Lewin, John Rahn, . . .) as well as the European tradition (Pierre Barbaud, Iannis Xenakis, Anatol Vieru, Michel Philippot, Guerino Mazzola, . . .),[3] the combinatorial character of the equal-tempered system is formally described by the algebraic model of the cyclic group $\mathbb{Z}/n\mathbb{Z}$ of order n, where n denotes the number of equal divisions of the octave. It is also common usage, at least in European-oriented computational musicology (Riotte and Mesnage 2006), to make use of the circular representation in order to geometrically capture the algebraic properties of this group within a computer-aided environment. A given chord of m distinct notes (modulo the octave) can therefore be represented from a geometric point of view as an m-polygon inscribed in a circle. To each chord one can associate a sequence of m integers counting the successive intervals in the chord (interval structure). Such a structure is an algebraic invariant, enabling one to identify in a unique way a given chord up to transpositions, as thoroughly studied by Anatol Vieru, whose catalogue of chords is provided by these means (Vieru 1980).

Periodic rhythmic patterns can also be represented via the N-CERCLE object. Example 6 shows two rhythmic patterns that are particularly interesting from a tiling perspective. They are called "3-asymmetric" in Hall and Klingsberger's terminology (Hall and Klingsberger 2006). From a group-theoretical point of view they are what mathematical music theorists usually call "self-inverse partitioning" musical structures. In fact they are "inversionally" related, meaning that they are basically the same orbit under the action of the dihedral group on the twelve-tone system. Moreover, the twelve-tone system can be partitioned into a disjoint union of transposition classes of the two structures. We show their circular representation (by means of the N-CERCLE object), their intervallic structure (provided by the *N*-STRUCTURE function), their interpretation as chords, and one possible rhythmic interpretation via the function C2RHYTHM, which maps the circular representation onto a corresponding rhythmic pattern (with the choice of a minimal rhythmic value corresponding to the eight note).



EXAMPLE 6: TWO MUSICAL STRUCTURES REPRESENTED AS 4-POLYGONS INSCRIBED IN A CIRCLE AND INTERPRETED AS INVERSIONALLY-RELATED CHORDS AND 3-ASYMMETRIC RHYTHMIC PATTERNS

2. A First Computational Approach to the Construction of Tiling Rhythmic Canons: Constraint Programming

A rhythmic canon is a polyphony of the same rhythm translated in time. It is defined by two rhythmic patterns: the inner one (R) that is the ground voice (a periodic rhythm) and the outer one (S) made by the entries of each voice. A tiling rhythmic canon is a rhythmic canon where at each pulsation one and only one voice attacks a note. Because of the simplicity of the definition of a tiling rhythmic canon, one might imagine that modeling this structure as a CSP (constraint satisfaction problem) is a very natural approach. A CSP is given by a triplet (X, D, C) where $X=\{x_1,\ldots,x_n\}$ is a finite set of variables, D a set of finite domains, one for each variable (we denote $\delta(x_i)$ the domain of the variable x_i), and C a set of constraints of the form (Z,R) with Z being a subset of X and R a subset of the Cartesian product of their domains:

$$\delta(x_1)\times\cdots\times\delta(x_r)\supseteq R(x_1,\ldots,x_r).$$

Values $v_{x_1}\in\delta(x_1),\ldots,v_{x_r}\in\delta(x_r)$ satisfy a constraint $R(x_1,\ldots,x_r)$ if

$$(v_{x_1},\ldots,v_{x_r})\in R(x_1,\ldots,x_r).$$

An instantiation I is a set of values $(v_{z_1},\ldots,v_{z_r})$ assigned to a subset $Z\subseteq X$. If the inclusion of Z in X is strict, we say that I is a *partial instantiation*. The instantiation I is consistent if, for all constraints C_i containing variables in Z, the values in I satisfy C_i. A solution is an instantiation consistent with $Z=X$. As an example, suppose that we are trying to find a tiling rhythmic canon of four voices with period sixteen. The CSP that models this problem is given by:

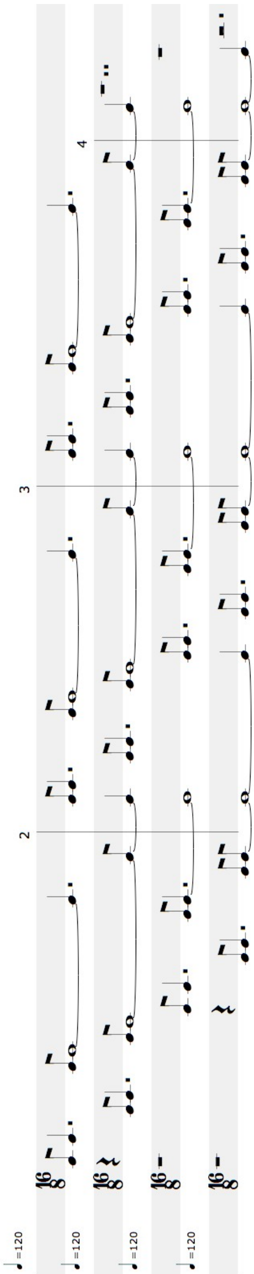
a. Variables: $R=\{x_1,\ldots,x_6\}\cup S=\{y_1,y_2,y_3,y_4\}$
b. Domains: $\delta(x_i)=(0\ldots15)=\delta(y_j)$
c. Constraints:

$$(x_1,\ldots,x_6)\otimes(y1,y2,y3,y4)=\text{ a permutation of }(0,\ldots,15)$$

In order to remove duplicates by translation we add two new constraints: $x_1=0$ and $y_1=0$.

Once the CSP is defined, it is given as an input to a constraint solver.[4] We obtain a solution $R=\{0,1,4,5\}$ and $S=\{0,2,8,10\}$ that corresponds to the tiling canon in Example 7. Examples in this paper take the eighth-note as the basic pulse.

It is easy to generalize this example as a function taking two parameters p and v specifying respectively the period and the number



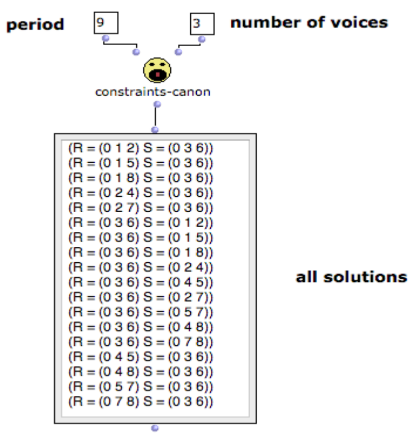
EXAMPLE 7: A TILING RHYTHMIC CANON OF FOUR VOICES OF PERIODICITY SIXTEEN, OBTAINED BY MODELING THE CYCLIC GROUP FACTORIZATION AS A CONSTRAINT SATISFACTION PROBLEM

of voices. An advantage of constraint programming is that it allows searching exhaustively all the solutions for a given period. For instance, for p=9 and v=3 there are eighteen canons, as shown in Example 8.

The main problem with constraint programming is the time of computation, which can hinder applications in the factorization of cyclic groups with larger periods (as, for example, 72). It is also difficult to add new constraints to the CSP without changing the performance time (as in the families of Vuza and augmented canons). For these reasons we need to know more about the problem in order to find more interesting solutions.

3. Tiling Rhythmic Canons and Milton Babbitt's Combinatoriality

As mentioned, the circular representation provides one of the most practical visualization tools for describing the inner symmetries of different kinds of musical structures. In our attempt to implement some computational properties of set theory and transformational analysis, we initially focused on several families of chord structures having specific symmetries up to transposition, inversion, and complementation

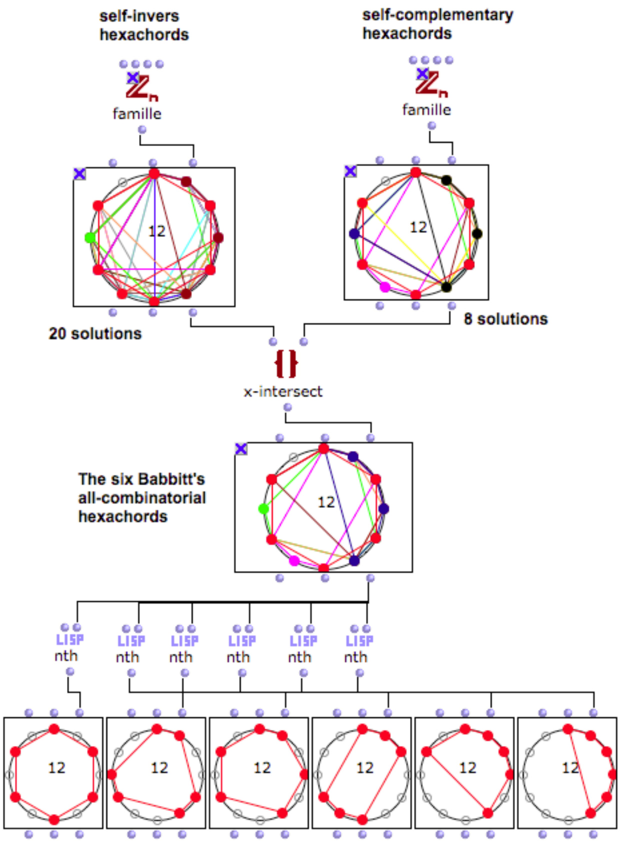


EXAMPLE 8: THE COMPLETE CATALOGUE OF SOLUTIONS OF THE FACTORIZATION PROBLEM OF THE CYCLIC GROUP OF ORDER NINE IN TWO SUBSETS WITH CARDINALITY THREE

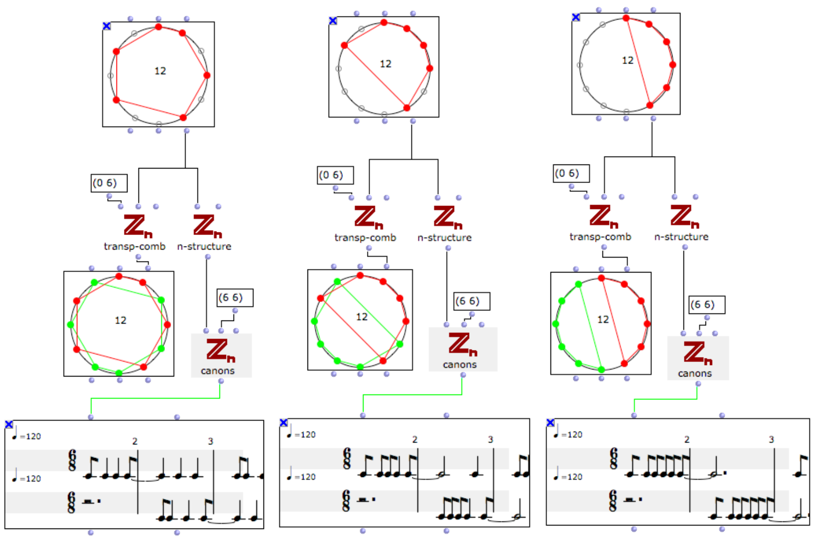
(or a combination of the previous properties). These not only correspond to well-known cases in mathematical music theory (Halsey and Hewitt 1978; Vieru 1980) but also have a natural interpretation in terms of tiling rhythmic canons once the pitch content is mapped into the rhythmic domain. As an elementary example let us consider the case of Milton Babbitt's all-combinatorial hexachords. By definition these structures have the property of being at the same time self-complementary (i.e., invariant under complementation) and self-inverse (i.e., invariant under inversion). The OpenMusic patch in Example 9 shows how the six all-combinatorial hexachords are obtained via graphical programming, starting from the two collections of, respectively, twenty self-inverse and eight self-complementary chords.

Since each of the six all-combinatorial hexachords tiles the chromatic space by transposition of a given interval (minor second, major second, minor third, or tritone), it is easy to construct tiling rhythmic canons in two voices having as inner rhythm one of the six all-combinatorial hexachords (mapped in the time domain) and as the outer voice one of the subsets obtained by the previous transposition values. Notice that among the six previous solutions, three, in fact, correspond to rhythmic patterns having a period that is strictly less than twelve. In other words, they have transpositional symmetry, allowing the reduction of the rhythmic space to be tiled. From the theory of Hajós groups we know that the three non-periodic inner rhythms will generate the tiling in accordance with an outer rhythm corresponding to a periodic subset of $\mathbb{Z}/12\mathbb{Z}$ of cardinality two. Since the only possibility for such a subset to be periodic is as a subgroup of order two, we deduce that the corresponding tiling canons will have two voices entering regularly with a distance equal to six units. The patch in Example 10 calculates and visualizes the three tiling rhythmic canons corresponding to the non-periodic all-combinatorial hexachords. The function TRANSP-COMB corresponds to the set-theoretical operation of "transpositional combination" (Cohn 1986), visualizing the tiling process by means of the circular representation.

The previous example is paradigmatic of a series of examples of tiling rhythmic canons that can be easily calculated since the voices of the canons enter in a regular way, according to the property of one of the factors of the decomposition of $\mathbb{Z}/n\mathbb{Z}$ being a subgroup of the cyclic group. As we already mentioned, this corresponds to the case of k-asymmetry, a property generalizing Simha Arom and Marc Chemillier's odditivity (Chemillier 2002). It is easy to show that the traditional definition of a k-asymmetric rhythmic pattern A is equivalent to the condition that $\mathbb{Z}/n\mathbb{Z}=A+B$, where B is the subgroup



EXAMPLE 9: ENUMERATION AND VISUALISATION OF MILTON BABBITT'S ALL-COMBINATORIAL HEXACHORDS STARTING FROM THE COLLECTION OF THE TWENTY SELF-INVERSE AND THE EIGHT SELF-COMPLEMENTARY HEXACHORDS

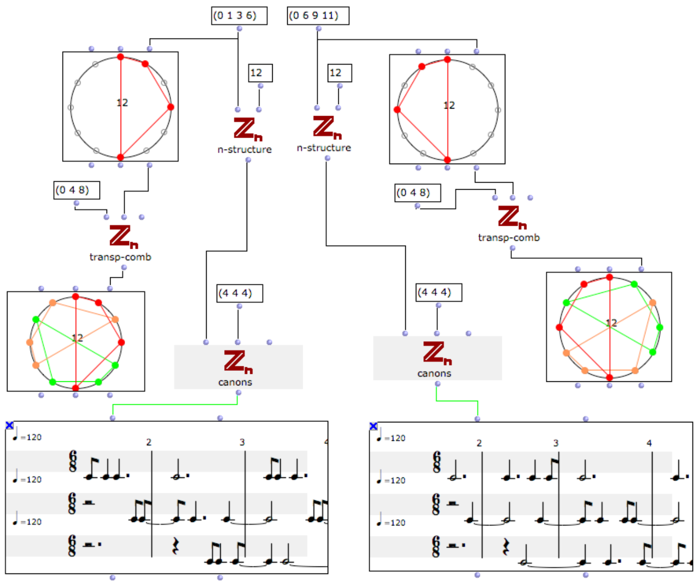


EXAMPLE 10: AN OPENMUSIC PATCH SHOWING HOW TO OBTAIN THE THREE RHYTHMIC TILING CANONS CORRESPONDING TO NON-PERIODIC, ALL-COMBINATORIAL HEXACHORDS

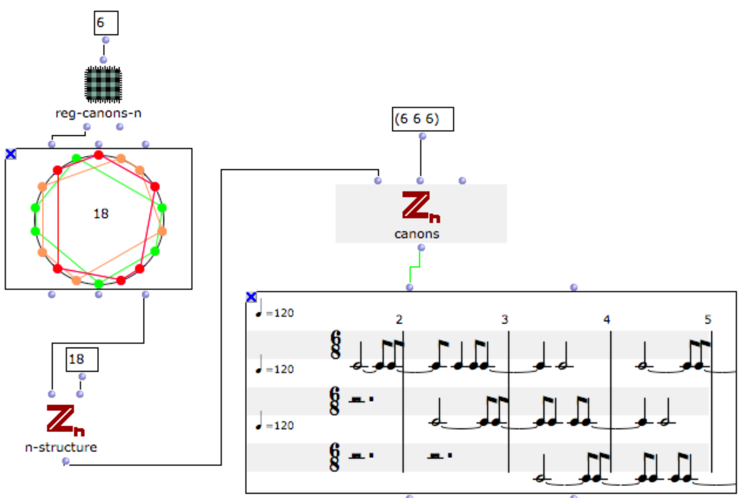
of $\mathbb{Z}/n\mathbb{Z}$ of cardinality equal to k. Example 11 shows how to generate two tiling rhythmic canons starting from the two inversionally-related 3-asymmetric patterns of Example 6.

Tiling rhythmic canons of this type are easy to construct. In fact, one can show that the inner rhythmic pattern of a k-asymmetric rhythm always reduces to the entire cyclic group $\mathbb{Z}/m\mathbb{Z}$ when it is reduced mod m, where $m=12/k$. This means that a k-asymmetric rhythm is generated by a permutation of $\{0,1,\ldots,m-1\}$. The patch in Example 12 shows how to construct all tiling rhythmic canons starting from a k-asymmetric pattern of cardinality equal to six. In this example $k=3$, which means that the pattern tiles cardinality six within the cyclic group $\mathbb{Z}/18\mathbb{Z}$.

Note that by fixing the cardinality m of the inner rhythm (which is equal to six in the previous case), the order n of the underlying cyclic group being tiled is always a multiple of m, which immediately provides the information on the number n/m of voices of the canon (hence the order of the asymmetry of the initial pattern). In this way, it is easy to construct canons with big periods, eventually corresponding



EXAMPLE 11: TWO TILING RHYTHMIC CANONS BASED ON INVERSIONALLY-RELATED, 3-ASYMMETRIC PATTERNS

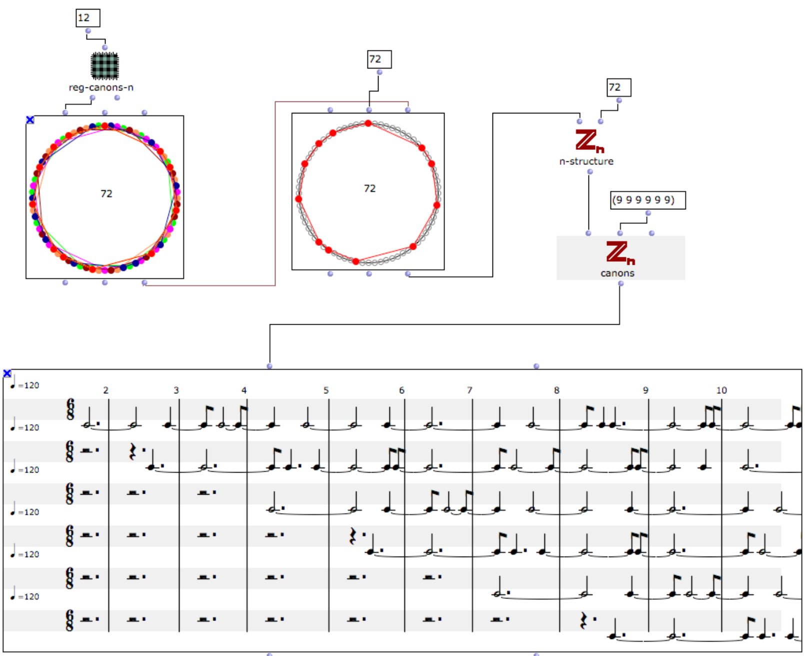


EXAMPLE 12: GENERATION OF A TILING RHYTHMIC CANON OF PERIOD EIGHTEEN STARTING FROM THE 6-ELEMENT PERMUTATION (0,5,1,3,4,2) INTERPRETED AS AN INNER RHYTHMIC PATTERN IN $\mathbb{Z}/18\mathbb{Z}$

to one of the values provided by the theory of non-Hajós groups. For example, we can show how to construct a tiling rhythmic canon in six voices starting from a 6-asymmetric inner rhythmic pattern that eventually provides a tiling of the non-Hajós group $\mathbb{Z}/72\mathbb{Z}$ (see Example 13). Obviously, this is not a Vuza Canon, since one of the factors corresponds to a subgroup of $\mathbb{Z}/72\mathbb{Z}$.

4. Deriving Tiling Rhythmic Canons from the Theory of Cyclotomic Polynomials

The use of polynomials to represent the factorization of a cyclic group into subsets goes back to the Minkowski-Hajós problem.[5] As originally observed by Rédei (1947), a decomposition of a cyclic group $\mathbb{Z}/n\mathbb{Z}$ into a direct sum $A \oplus B$ of two subsets A and B, can be equivalently



EXAMPLE 13: A TILING RHYTHMIC CANON OF PERIOD 72 CONSTRUCTED FROM THE 12-ELEMENT PERMUTATION (0,10,1,8,4,2,7,9,5,11,3,6) INTERPRETED AS AN INNER RHYTHMIC PATTERN IN $\mathbb{Z}/72\mathbb{Z}$

expressed in polynomial terms. By putting $A(x)=\sum x^{\alpha}$, where $\alpha\in A$, the above equation becomes a relation between polynomials with coefficients being either 0 or 1, which will be referred to as "0-1 polynomials." The factorization condition is expressed by the fact that $A(x)\times B(x)\equiv 1+x+x^2+\cdots+x^{n-1}\ (\mathrm{mod}\ x^n-1)$. Factors of the polynomial $\Delta_n(x)=1+x+x^2+\cdots+x^{n-1}$, especially those with 0-1 coefficients, are thus of paramount importance, which is the main reason to use the theory of cyclotomic polynomials (Amiot, *et al.* 2005). More precisely, if $A(x)\times B(x)\equiv 1+x+x^2+\cdots+x^{n-1}\ (\mathrm{mod}\ x^n-1)$, then for all $d|n$ (with $d\neq 1$), Φ_d is a divisor of either $A(x)$ or $B(x)$, where Φ_d is the d^{th} cyclotomic polynomial.

In the MathTools environment we have collected a number of tools enabling one to obtain tiling rhythmic canons by using cyclotomic polynomials. With exception of some special cases that can be found in the "tiling of the line" literature (and which we discussed in Andreatta, *et al.* 2002), it is not known whether a given rhythmic motif enables making a canon unless one is able to exhibit such a canon. Since the establishment of the two Coven and Meyerowitz conditions (Coven and Meyerowitz 1998), there is a useful, sufficient criterion dealing with the cyclotomic factors. Let $A(x)$ be the 0-1 polynomial associated with an inner rhythm A. We define the two following sets:[6]

$R_A=\{d\in Z, d>0, \Phi_d|A(x)\}$, and
$S_A=\{d\in R_A : d=p^{\alpha} \text{ where } p \text{ is prime and } \alpha\in N\}$.

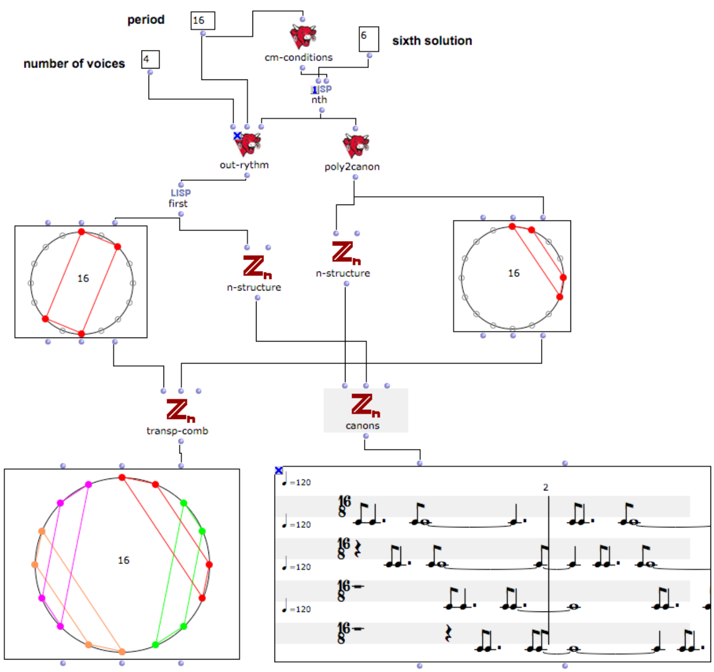
The two Coven-Meyerowitz conditions, called (T1) and (T2), may be expressed in the following way:

(T1) $A(1)=\prod p$, where the product runs over all $p^{\alpha}\in S_A$, and
(T2) If $p^{\alpha}, q^{\beta}\ldots\in S_A \Rightarrow (p^{\alpha}\times q^{\beta}\ldots)\in R_A$.

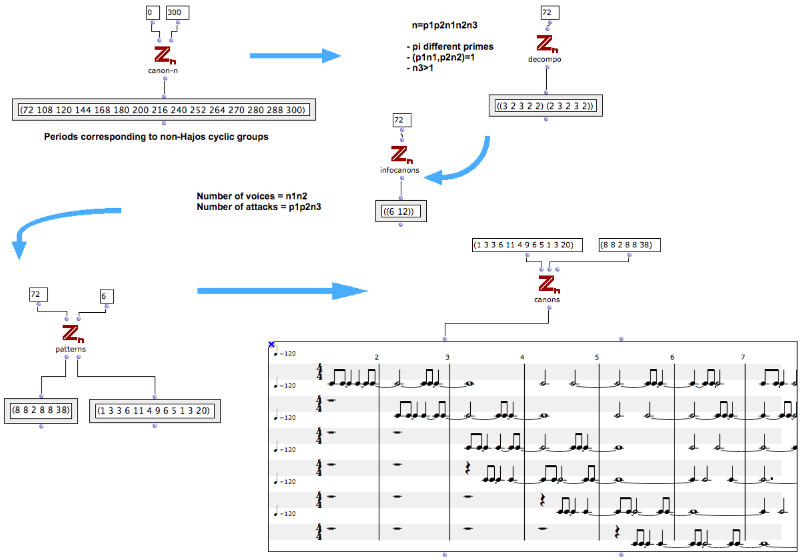
Tiling rhythmic canons constructed from a 0-1 polynomial $A(x)$ that has the properties (T1) and (T2) are called "Cyclotomic Canons." As an example, we show how to construct the same canon as in Example 7 with this approach. By choosing as input the value 16, the function CM-CONDITIONS gives all the possible 0-1 polynomials $A(x)$ associated to subsets of $\mathbb{Z}/16\mathbb{Z}$ having both (T1) and (T2) properties. The sixth solution corresponds to the 0-1 polynomial $A(x)=1+x+x^4+x^5$ obtained by the product of the two cyclotomic polynomials Φ_2 and Φ_8. The OUT-RYTHM function constructs the polynomial $B(x)$ which assures the factorization of $1+x+\cdots+x^{18}$ as $A(x)\times B(x)$, which provides the tiling visualized in the circular representation or as a rhythmic canon, which, in turn, is the same canon obtained through CSP, as we intended (Example 14).

5. A Step-by-Step Description of Vuza Canon Constructions in the OpenMusic Visual Programming Language

Our implementation of the algorithm proposed by Vuza in his series of articles published in *Perspectives of New Music* (Vuza 1991–1993) was the first attempt at studying some computational aspects explicitly raised by him. Before describing the first exhaustive catalogue of solutions that we have obtained in *OpenMusic*, let us briefly provide a general overview of the main tools that we implemented in order to obtain Vuza canons. These are shown in Example 15.



Example 14: A cyclotomic canon of period 16 obtained from a 0-1 polynomial satisfying the two Coven-Meyerowitz conditions



EXAMPLE 15: OVERVIEW OF THE MAIN TOOLS IMPLEMENTED IN OPENMUSIC IN ORDER TO OBTAIN VUZA CANONS OF A GIVEN PERIOD

First of all, as we know from the theory of non-Hajós groups, Vuza Canons only exist for some given period n that does not have the property of being:

- a power p^α of a prime number,
- a product p^2q^2 of the square of two distinct prime numbers,
- a product $p^\alpha q$ of the power of a prime number by a different prime number,
- a product p^2qr of the power of a prime number by two other distinct prime numbers, and
- a product $pqrs$ of four distinct prime numbers.

The function CANON-N calculates all periods corresponding to non-Hajós cyclic groups within a given interval (in the previous example, the periods n are constrained between 0 and 300). Every such a period n is decomposable as a product of five numbers $p_1p_2n_1n_2n_3$ where:

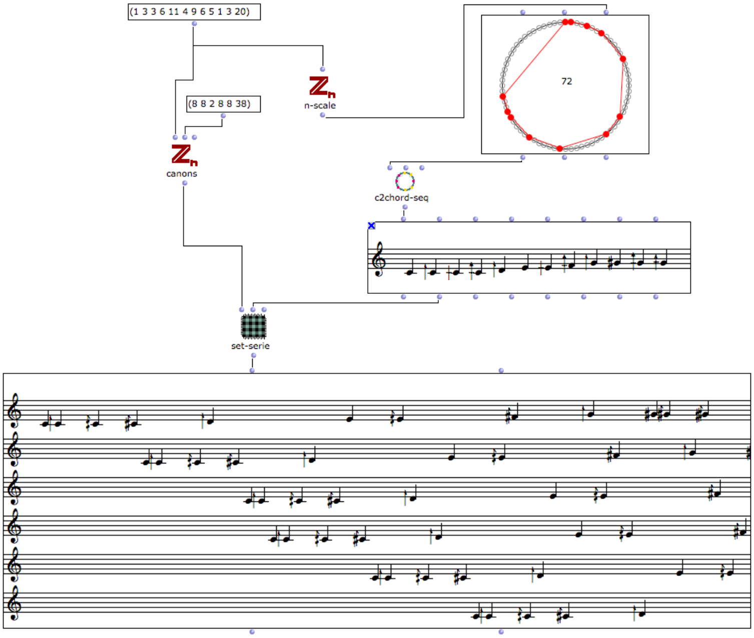
i. p_1 and p_2 are two distinct primes,
ii. $< p_1n_1, p_2n_2 > = 1$ (i.e., p_1n_1 is relatively prime with p_1n_2), and
iii. n_1 is an integer greater than 1.

The decomposition is obtained by the function DECOMPO, which takes as input an integer n corresponding to the period of a non-Hajós cyclic group. Note, that the decomposition is not necessarily unique, as the example of 72 shows. Nevertheless, the resulting canons may have the same structural properties in terms of number of voices and number of attacks within each voice (these two values being respectively equal to n_1n_2 and $p_1p_2n_3$). These structural data are provided by the function INFOCANON. By choosing a valid number of voices, the factorization of the group into two non-periodic factors is obtained through the function PATTERNS, and these sets will finally be provided as input for the function CANONS in order to construct the rhythmic grid of the required Vuza canon. The representation of the canon in the previous example makes use of the POLY editor in rhythmic mode. The following example (Example 16) shows two different representations of the same canon by including some information on the pitch content. The first one makes use of the MULTI-SEQ editor, in which the notes are represented by using proportional durations, whereas the second one uses the same POLY editor but in a pitch-content mode. In order to show that we are not limited to twelve-tempered equal systems, we map the information of the inner rhythmic pattern into the microtonal space $\mathbb{Z}/72\mathbb{Z}$ corresponding to the twelfth-tone division of the octave.[7]

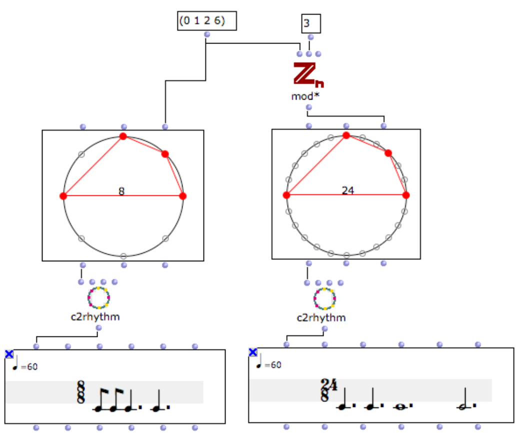
6. Canons by Augmentation

In this section we will present a generalization leading to canons with augmented voices.[8] We will, therefore, explore a new symmetry of $\mathbb{Z}/n\mathbb{Z}$ under the group of affine transforms. In this new context each symmetry $[a,t](x):=ax+t$ consists of an augmentation with factor a and a translation with summand t. Example 17 shows a rhythmic pattern of period 8 that has been augmented by a factor of 3.

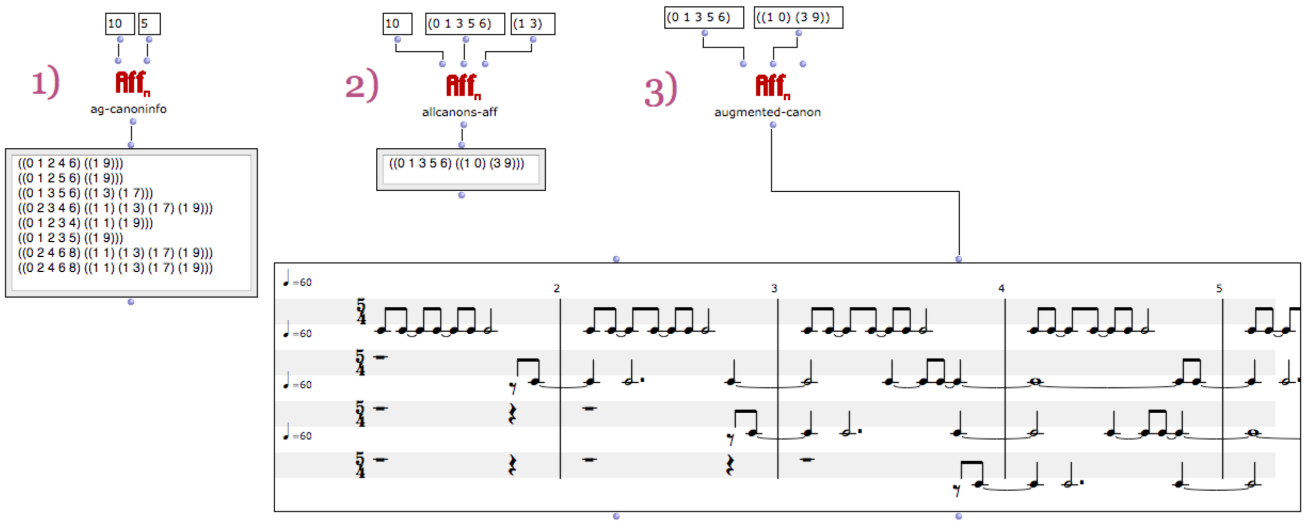
Let us provide an overview of the main tools that we implemented, shown in Example 18. Three steps can be identified. In step one the user starts by specifying a period p (10, in our example) and the number of attacks compounding the rhythmic pattern (5). The function AG-CANONINFO provides the list of rhythmic patterns of length 5, together with the multiplicative factors, which need to be applied on them in order to build tiling canons in which each voice is an augmentation of the original pattern according to these multiplicative factors. In Example 18 the solution ((0 1 3 5 6) ((1 3) (1 7))) means that the 5-element rhythmic pattern (0 1 3 5 6) can generate a tiling



EXAMPLE 16: A MICROTONAL CANON OBTAINED FROM A VUZA CANON OF ORDER 72



EXAMPLE 17: AUGMENTATION OF A RHYTHM



EXAMPLE 18: OVERVIEW OF THE MAIN TOOLS IMPLEMENTED IN OPENMUSIC IN ORDER TO OBTAIN AUGMENTED CANONS

canon both via multiplications by 1 (the original pattern) and by 5 or multiplications by 1 and 7.

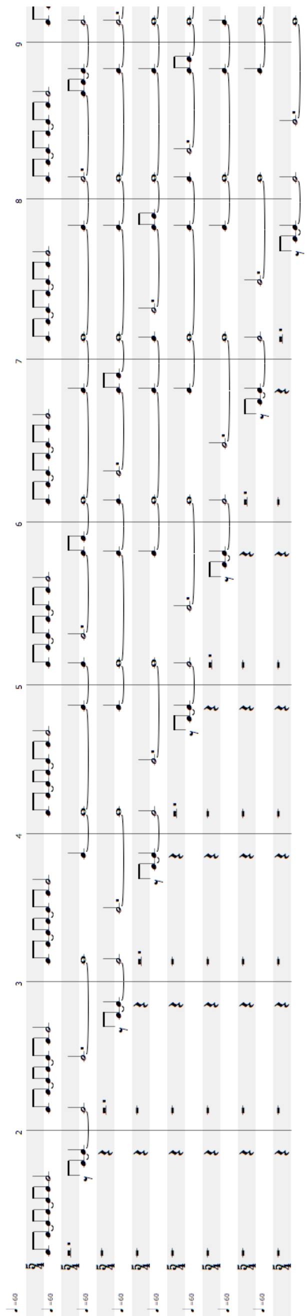
At this moment the user must choose the rhythm and the multiplicative factors. Once the choice is made, these two parameters are given to the function ALLCANONS-AFF (step two). In our example we provide to this function the pattern (0 1 3 5 6) and the factors (1 3). The function ALLCANONS-AFF provides, for the original pattern, the list of couples $(a_i\ b_i)$, where a_i are the multiplicative factors and b_i are the translation factors. For example, the solution ((0 1 3 5 6) ((1 0) (3 9))) means that the 5-element rhythmic pattern (0 1 3 5 6) can generate a tiling canon via multiplications by 1 and translation by 0 (i.e., the original pattern) and multiplication by 3 and translations by 9 units.

Finally, the function AUGMENTED-CANON uses this information in order to concretely build a tiling rhythmic canon in which all voices are augmentations of the original rhythmic pattern. The period of the augmented canon is given by the period of the original pattern times the least common multiple of the factors ($10\times3=30$ in Example 18). The voices augmented by 3 are repeated 3 times and translated by 9, 19, and 29

Note that for the pattern (0 1 3 5 6) we have another solution with multiplicative factors (1 7) (see Example 19). In this case, the result of the function ALLCANONS-AFF is ((0 1 3 5 6) ((1 0) (7 7))) which means that the pattern is completed by 7 voices augmented by a factor 7 and translated by 7, 17, 27, 37, 47, 57, and 67. The period of this canon is $10\times7=70$.

7. Conclusions

The problem of constructing tiling rhythmic canons shows the usefulness of a mathematical approach to the formalization of these musical structures. This paper shows, in a practical way, how to construct several families of canons in OpenMusic. The reader may consider our references for a more mathematical description of these algorithms.



EXAMPLE 19: AUGMENTED CANON WITH FACTOR SEVEN

Notes

1. For more information on OpenMusic, the reader may consult Agon (1998) and Assayag, *et al.* (1999). For compositional uses of OpenMusic see the two volumes published in the series Musique/Sciences (Agon, *et al.* 2006; Bresson, *et al.* 2008).
2. For a short presentation of the "mathemusical" perspective in mathematical music theory, see Andreatta's contribution in the present issue of *Perspectives of New Music*. A more general discussion is contained in Andreatta (2010).
3. For a description of emergence of group-theoretical structures in both European and American traditions, see Andreatta (2003). A historical presentation of classification models for chords within the equal-tempered system from the European perspective is given by Luigi Verdi (2008).
4. See Truchet and Assayag (2011) for a review of different solvers and their musical applications.
5. See Andreatta's contribution to the present issue of *Perspectives* for an historical presentation of Minkowski's original conjecture on the tiling of the Euclidean space by unit cubes and Hajós's solution in terms of group factorizations.
6. In the presentation of Coven and Meyerowitz conditions, we follow the simplified notation proposed by Giulia Fidanza (2007). For a more detailed discussion of these conditions and the role they are playing with respect to Fuglede's spectral conjecture, see the contribution by Emmanuel Amiot in the present issue of *Perspectives.*
7. For the modeling and implementation of microtonal structures in OpenMusic, see Bancquart (2008).
8. These canons are sometimes referred to as "Noll Canons," stressing the crucial contribution by Thomas Noll to the development of the theory. See also Noll, *et al.* (2001).

References

Agon, Carlos. 1998. "OpenMusic: Un Langage Visuel pour la Composition Assistée par Ordinateur." Ph.D. thesis, Université de Paris VI.

———, Gérard Assayag, and Jean Bresson. 2006. *The OM Composer's Book 1*. Musique/Sciences. Sampzon: Ircam/Delatour France.

Amiot, Emmanuel, Moreno Andreatta, and Carlos Agon. 2005. "Tiling the (Musical) Line with Polynomials: Some Theoretical and Implementational Aspects." In *International Computer Music Conference Proceedings: vol. 2005*. Ann Arbor, MI: Scholarly Publishing Office, University of Michigan Library.

Andreatta, Moreno. 1999. "La Théorie Mathématique de la musique de Guerino Mazzola et les canons rythmiques." Master's thesis, Université de Paris IV/EHESS/IRCAM.

———. 2003. "Méthodes algébriques en musique et musicologie du XXe siècle: aspects théoriques, analytiques et compositionnels." PhD diss., IRCAM/ EHESS.

———. 2004. "On Group-Theoretical Methods Applied to Music: Some Compositional and Implementational Aspects." In *Perspectives in Mathematical and Computational Music Theory*, edited by G. Mazzola, T. Noll, and E. Puebla, 122–162. Electronic Publishing Osnabrück.

——— 2010. *Mathematica est exercitium musicae: la recherche 'mathémusicale' et ses interactions avec les autres disciplines*. Habilitation Thesis, Institut de Recherche Mathématique Avancée, University of Strasbourg. Available at http://recherche.ircam.fr/equipes/remus/moreno/TexteHDR.pdf.

Andreatta, Moreno, Agon, Carlos & Amiot, Emmanuel. 2002. "Tiling Problems in Music Composition: Theory and Implementation," In *International Computer Music Conference Proceedings: vol. 2002*, 156–163. Ann Arbor, MI: Scholarly Publishing Office, University of Michigan Library.

Andreatta, Moreno, Carlos Agon, Thomas Noll, and Emmanuel Amiot. 2006. "Towards Pedagogability of Mathematical Music Theory: Algebraic Models and Tiling Problems in Computer-Aided Composition." In *BRIDGES 2006 Conference Proceedings*, edited by

R. Sarhangi and J. Sharp, 277–284. London: Print Solutions Partnership.

Andreatta, Moreno, and Carlos Agon. 2009. "Guest Editors' Foreward." *Journal of Mathematics and Music* 3/2: 63–70.

Assayag, Gérard, Camilo Rueda, Mickael Laurson, Carlos Agon, and Olivier Delerue. 1999. "Computer Assisted Composition at IRCAM: From PatchWork to OpenMusic." *Computer Music Journal* 23/3: 59–72.

Bancquart Alain, Moreno Andreatta, and Carlos Agon. 2008. "Microtonal Composition," in *The OM Composer's Book 2*, ed. Jean Bresson, Carlos Agon, and Gérard Assayag, 279–302. Musique/Sciences. Sampzon: IRCAM/Delatour France.

Bloch, Georges. 2006. "Vuza Canons into the Museum." In *The OM Composer's Book* 1, edited by C. Agon, G. Assayag, and J. Bresson. Collection Musique/Sciences. Sampzon: Ircam-Delatour France.

Bresson Jean, Carlos Agon, Gérard Assayag. 2008. *The OM Composer's Book 2.* Musique/Sciences. Sampzon: Ircam/Delatour France.

Chemillier, Marc. 2002. "Ethnomusicology, Ethnomathematics. The Logic Underlying Orally Transmitted Artistic Practices." in *Mathematics and Music: A Diderot Mathematical Forum*, edited by G. Assayag, H. G. Feichtinger, and J .F. Rodrigues, 161–183. Berlin: Springer Verlag.

———. 2007. *Les mathématiques naturelles.* Paris: Odile Jacob.

Cohn, Richard. 1986. "Transpositional Combination in Twentieth-Century Music." PhD thesis, University of Rochester, Eastmann School of Music.

Coven, Ethan, and Aaron Meyerowitz. 1999. "Tiling the Integers with Translates of One Finite Set." *Journal of Algebra* 212/1: 161–174.

Fidanza, Giulia. 2007. "Canoni ritmici a mosaico." PhD thesis, Università degli Studi di Pisa.

Fripertinger, Harald. 2001. "Enumeration of non-isomorphic canons." *Tatra Mountains Mathematical Publications* 23: 47–57

Hall, Rachel, and Paul Klinsberg. 2006. "Asymmetric Rhythms and Tiling Canons." *American Mathematical Monthly* 113/10: 887—896.

Halsey, David, and Edwin Hewitt. 1978. "Eine gruppentheoretische Methode in der Musik-theorie," *Jahresbericht. der Deutschen Mathematiker-Vereinigung* 80: 151-207.

Moreno, Andreatta, Thomas Noll, Carlos Agon, and Gérard Assayag. 2001. "The Geometrical Groove: Rhythmic Canons Between Theory, Implementation and Musical Experiments", In *Actes des Journées d'Informatique Musicale, Bourges*: 93–98.

Rédei, Laszlo. 1947. "Zwei Lückensätze Über Polynome in Endlichen Primkörpern mit Anwendung auf die endlichen Abelschen Gruppen und die Gaussischen Summen." *Acta Mathematica* 79/1: 273–290.

Riotte, André, and Marcel Mesnage. 2006. *Formalismes et modèles musicaux: un recueil de texts, 1963–1998*. Musique/Sciences. Sampzon: IRCAM/Delatour France.

Truchet, Charlotte, and Gérard Assayag, eds. 2011. Constraint Programming in Music. London: ISTE/Wiley.

Verdi, Luigi. 2008. "The History of Set Theory from a European Point of View." In *Proceedings of the Symposium Around Set Theory: A French/American Musicological Meeting, IRCAM, October 15–16, 2003*, ed. Moreno Andreatta, Jean-Michel Bardez, and John Rahn. Musique/Sciences. Sampzon: IRCAM/Delatour France.

in M. Andreatta, J.-M. Bardez & J. Rahn (eds), *Around Set Theory. A French/American Musicological Meeting*, "Musique/Sciences Series". Sampzon: Ircam/Delatour France.

Vieru, Anatol. 1980. *Cartea modurilor 1*. Bucarest: Ed. Muzicală. (English translation and enlarged version published as *The Book of Modes*, Bucarest: Editura Muzicală, 1993).

Vuza, Dan Tudor. 1985. "Sur le rythme périodique." *Revue Roumaine de Linguistique-Cahiers de linguistique Théorique et Appliquée* 23/1: 73–103.

———. 1988. "Some Mathematical Aspects of David Lewin's Book *Generalized Musical Intervals and Transformations.*" *Perspectives of New Music* 26/1: 258–287.

———. 1991–1993. "Supplementary Sets and Regular Complementary Unending Canons." *Perspectives of New Music* 29/2: 22–49 (Part I); 30/1: 184–207 (Part II); 30/2: 102–125 (Part III); 31/1: 270—305 (Part IV).