



HAL
open science

OMM: an OpenMusic library for Mathematical Morphology

Carlos Agon, Gonzalo Romero-García, Isabelle Bloch

► **To cite this version:**

Carlos Agon, Gonzalo Romero-García, Isabelle Bloch. OMM: an OpenMusic library for Mathematical Morphology. 2024 International Computer Music Conference 2024, Jul 2024, Seoul, South Korea. <hal-04914319>

HAL Id: hal-04914319

<https://hal.science/hal-04914319v1>

Submitted on 27 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

OMM: an OpenMusic library for Mathematical Morphology

Carlos Agon

Sorbonne Université,
CNRS, IRCAM, STMS,
Paris, France
agon@ircam.fr

Gonzalo Romero-García

Sorbonne Université,
CNRS, IRCAM, STMS,
Paris, France
romero@ircam.fr

Isabelle Bloch

Sorbonne Université,
CNRS, LIP6,
Paris, France
isabelle.bloch@sorbonne-universite.fr

ABSTRACT

This article introduces OMM, a new OpenMusic library based on Mathematical Morphology. After conducting several years of research on the potential usages of Mathematical Morphology operators in composition, musical analysis, and music education, we are convinced that it is now opportune to develop computer tools for music creation, analysis, and education based on Mathematical Morphology concepts. Our primary objective is to demonstrate the relevance of these techniques in the musical domain. In pursuit of this goal, we present one illustrative example based on rhythmical transformations. OMM represents an initiative aimed at bridging the gap between theoretical ideas and practical applications in the musical domain. This contribution reflects our commitment to democratizing access to advanced musical tools, thereby promoting a more inclusive musical landscape.

1. INTRODUCTION

This article introduces OMM, an OpenMusic [1] library which applies Mathematical Morphology to music. Mathematical Morphology (MM) is a domain at the intersection of mathematics and computer science. It was originally formulated in the mid-1960s in [2] and [3].

MM has been highly successful in the field of image processing. We argue that MM can also be relevant in the musical domain, without the need to represent music as an image. The paper [4] suggested for the first time a possible strategy to approach musical information research by combining tools and ideas belonging to MM. More recently, two doctoral theses [5] and [6] have improved this research domain proposing relevant representations both in the symbolic and signal processing domains.

We argue that the application of MM to the musical domain is advantageous, mainly because its algebraic characteristics. The fundamental idea is to define abstract operators for the generation and analysis of musical materials in general. Once these operators are defined, it is crucial to find suitable representations where these tools

This research is supported by European Research Council ERC-ADG-883313 REACH, and Agence Nationale de la Recherche ANR-19-CE33-0010 MERCI.

Copyright: ©2024 Carlos Agon et al. This is an open-access article distributed under the terms of the [Creative Commons Attribution License 3.0 Unported](https://creativecommons.org/licenses/by/3.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

can be applied in a musically relevant manner. OMM provides a graphical interface for connecting various mathematical morphology operators to musical objects. We present in this paper one specific application case concerning rhythms. Furthermore, OMM establishes the necessary conditions to apply mathematical morphology operators to new musical materials, thereby offering users the opportunity to pursue new musical research.

This article is organized as follows: in Section 2, we define the basics of MM¹, specifically its two fundamental operators: dilation and erosion. Then we show how these operators combine to define two other fundamental ones: opening and closing. These four operators, which may seem abstract initially, will gain meaning through their applications in the following sections. In Section 3, we introduce a first application domain: rhythmic canons. In this section, we employ what is known as binary morphology. Section 4 is dedicated to outlining the protocol necessary for extending applications to other types of representations, provided that they adhere to the conditions imposed by MM. We quickly showcase in this section an application concerning melodies by using what is known as grayscale morphology. The last section allow us to draw conclusions and envision other potential scenarios for future research.

2. MATHEMATICAL MORPHOLOGY

Lattice structures are at the basis of MM. We start by giving the definition of complete lattices (see [8] to know more about lattice theory).

[Poset : partially ordered set] A poset is a pair $\langle P, \leq \rangle$ where P is a set equipped with a binary relation \leq , called the order of P , with \leq being reflexive, antisymmetric and transitive.

We focus on a particular class of posets, namely lattices.

[Lattice] A poset $\langle L, \leq \rangle$ is a lattice if every pair of elements $x, y \in L$ has both a least upper bound and a greatest lower bound.

[Upper and lower bounds] Let $\langle L, \leq \rangle$ be an ordered set and $S \subseteq L$. Then

- $x \in L$ is a *lower bound* of S if $\forall y \in S, x \leq y$,
- $x \in L$ is the *greatest lower bound* of S if $\forall y \in S^\ell, x \geq y$.

when S^ℓ denotes the set of all the lower bounds of S .

¹ For a real introduction to MM, we recommend [7].

The (*least*) *upper bound* and the set of all upper bounds S^u are defined dually by reversing the order (using \geq instead of \leq).

[Complete lattice] A lattice $\langle L, \leq \rangle$ is complete if every subset of $S \subseteq L$ has both a least upper bound and a greatest lower bound, called supremum and infimum. A complete lattice is always bounded: there is a greatest element $\top \in L$ such that $\forall x \in L. x \leq \top$ and a least element $\perp \in L$ such that $\forall x \in L. \perp \leq x$.

A lattice is an algebraic structure $\langle L, \sqcup, \sqcap \rangle$ where the binary operation \sqcup is called the join and \sqcap the meet. The join $x \sqcup y$ is the least upper bound of the set $\{x, y\}$ and the meet $x \sqcap y$ is its greatest lower bound. We use the notation $\sqcup S$ (resp. $\sqcap S$) to compute the least upper bound (resp. greatest lower bound) of the set S .

Let (L, \leq) and (L', \leq') be any two arbitrary lattices, the two basic operations of MM, erosion and dilation, are defined as follows:

[Dilation] $\delta : L \rightarrow L'$ is a dilation if $\forall x_i \in L$,
 $\delta(\sqcup x_i) = \sqcup' \delta(x_i)$.

[Erosion] $\varepsilon : L \rightarrow L'$ is an erosion if $\forall x_i \in L$,
 $\varepsilon(\sqcap x_i) = \sqcap' \varepsilon(x_i)$.

In other words, an operator δ (resp. ε) is a dilation (resp. erosion) if it commutes with the supremum (resp. the infimum).

In this paper, we will define these two operators in a simplified framework called deterministic MM with structuring element. The main idea is to transform an element $A \in L$ using another element $B \in L$ called the structuring element. It is to note that in our case, the domain and codomain of erosion and dilation are the same lattice, i.e. $L = L'$.

We define algorithms for dilation and erosion using structuring element as follows:

-For dilation, we take the origin of the structuring element B , and translate it to occupy successively all positions within the space A . For each position, if B intersects A then this position is part of the dilation. We will note the dilation of A by B as $A \oplus B$.

-For erosion, we take the origin of the structuring element B , and translate it to occupy successively all positions within the space A . For each position, if B is completely included within A then this position is part of the erosion. We will note the erosion of A by B as $A \ominus B$.

The problem with these algorithms is that the notions of translation, intersection, inclusion and the origin of the structuring element depend entirely on the lattice that we use. In the following section, we will provide a specific example using what is called binary morphology.

Before concluding this section, we define two new operators based on erosion and dilation: opening and closing.

[Opening] $A \circ B = (A \ominus B) \oplus B$.

[Closing] $A \bullet B = (A \oplus B) \ominus B$.

In image processing it is said that opening removes narrow parts of the object and deforms convex corners of the object while closing fills narrow parts of the background and deforms concave corners of the object [7]. If we want to apply MM to music, it is necessary that we can make similar assertions about the application of these operators to musical structures.

The next section will provide specific examples of these four operators applied to rhythmical structures.

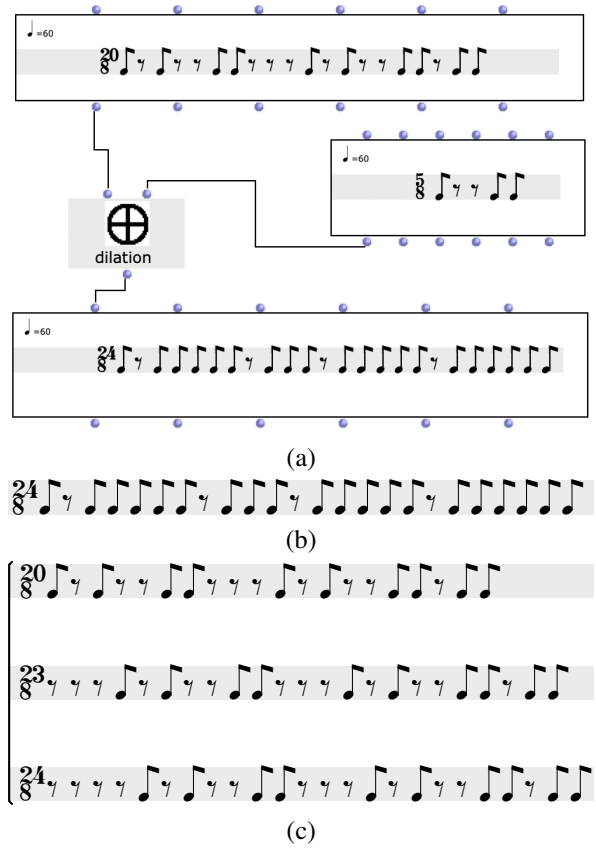


Figure 1. Dilation of one rhythm by another one.

3. RHYTHM

In this section, we work on what is called binary morphology. Binary morphology is often referred to as morphology on sets because it uses the lattice $(\mathcal{P}(E), \subseteq)$ with E a set and $\mathcal{P}(E)$ the power set of E .

The lattice that we will use here is defined by $(\mathcal{P}(\mathbb{Z}), \subseteq)$. Intuitively, we interpret a set X as a rhythmic pattern. For instance, $\{0, 2, 4, 5\}$ corresponds to the following rhythm $\frac{5}{8}$ $\text{♩} \text{ } \text{♩} \text{ } \text{♩} \text{ } \text{♩}$. In this example, we have taken the quarter note as the beat. There is a quarter note at position e for each element e in our set, and a rest otherwise.

In this context, we define the dilation of a rhythm A by a structuring rhythm B by the equation:

$A \oplus B = \{a + b \mid a \in A, b \in B\}$ with the usual addition in \mathbb{Z} .

Figure 1 (a) illustrates a program that performs the dilation computation of a rhythm $R_1 = \{0, 2, 5, 6, 10, 12, 15, 16, 18, 19\}$ by a structuring rhythm $R_2 = \{0, 3, 4\}$.

Analyzing the result in Figure 1 (b), we observe that it arises from the flattening of a rhythmic canon with internal rhythm R_1 and external rhythm R_2 . This canon is shown in Figure 1 (c). Dilation proves to be a straightforward method for constructing rhythmic canons, as defined in [9]. This observation is not surprising, given that we have defined binary dilation using the Minkowski addition in \mathbb{Z} . Rhythmic canons have been widely employed as techniques in various compositions, see [10].

Let us now see if we can attribute a rhythmic meaning to erosion. We define the erosion of a rhythm A by a structuring rhythm B by the equation:

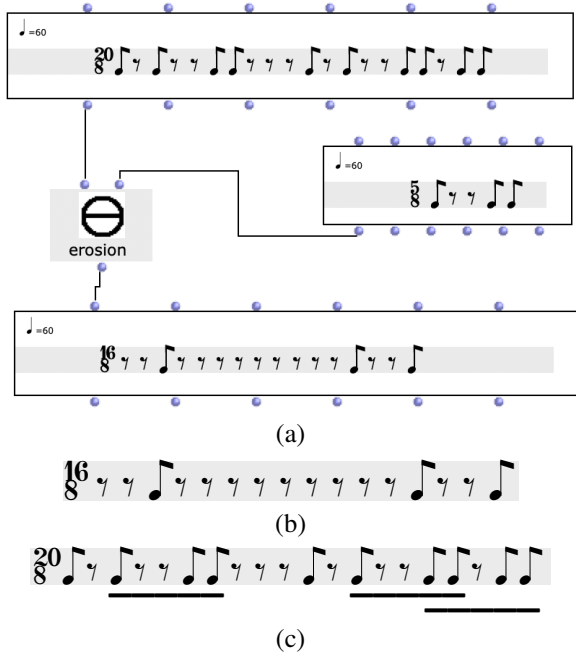


Figure 2. Erosion of one rhythm by another one.

$$A \ominus B = \{p \in A \mid B_p \subseteq X\}, \text{ where } B_p = \{x + p \mid x \in X\}.$$

Figure 2 (a) shows a program that calculates the erosion of the rhythm R_1 by the structuring rhythm R_2 .

Erosion computes a new rhythm, displayed in Figure 2 (b), wherein the attack points align with the starting moments in R_1 where one can find the rhythmic motif R_2 . From a musical analysis standpoint, this is comparable to seeking occurrences of R_2 within R_1 , similarly to searching for a word in a text (see Figure 2 (c)).

Let us now examine the rhythmic opening and closing operations. Figure 3 (a) illustrates a program that calculates the opening of R_1 by R_2 .

The result, depicted in Figure 3 (b), is equivalent to the flattening of the rhythmic canon in Figure 3 (c). This canon is obtained by taking as internal rhythm R_2 . The external rhythm is determined by the positions of R_2 within R_1 , that is, $R_1 \ominus R_2$.

The opening removes the attacks from the original rhythm that do not belong to any instantiation of the structuring rhythm. These can be thought of as a kind of rhythmic noise.

Figure 4 (a) illustrates a program that calculates the closing of R_1 by R_2 . Unlike opening, closing adds new attacks to the original rhythm (see Figure 4 (b)). This is due to the fact that the rhythmic space is first saturated by the construction of the rhythmic canon $R_1 \oplus R_2$ (Figure 4 (c)).

Experimentally, OMM incorporates additional operators specific to MM (e.g., hit-and-miss, area opening, area closing, see [7]). However, since their musical relevance is not evident, we have chosen not to present them in this article.

4. WORKING WITH OTHER LATTICES

OpenMusic is written in Common Lisp. For a user eager to extend OMM to other musical domains, the procedure consists in defining methods for the following generic functions:

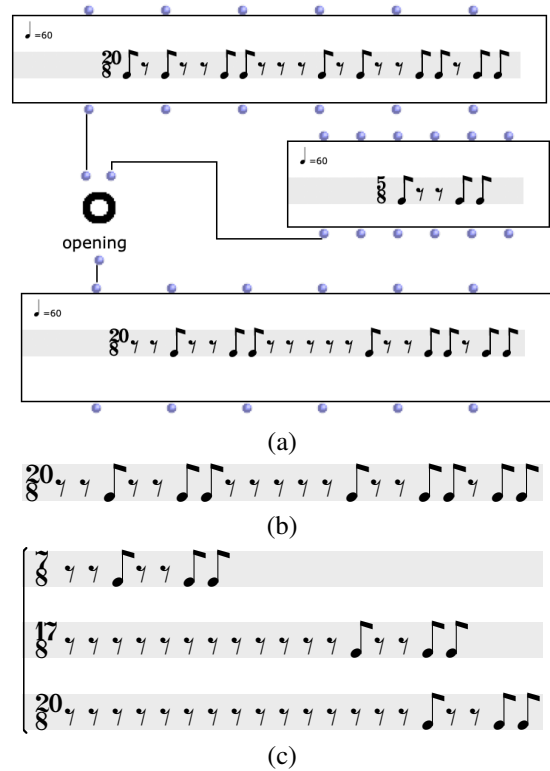


Figure 3. Opening of one rhythm by another one.

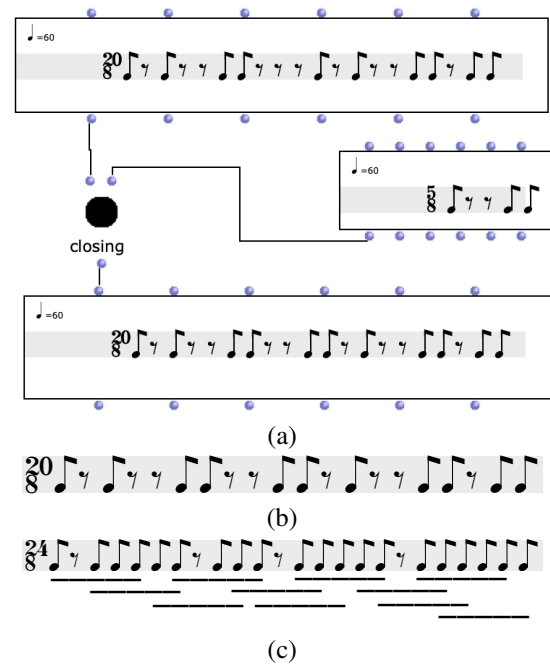


Figure 4. Closing of one rhythm by another one.

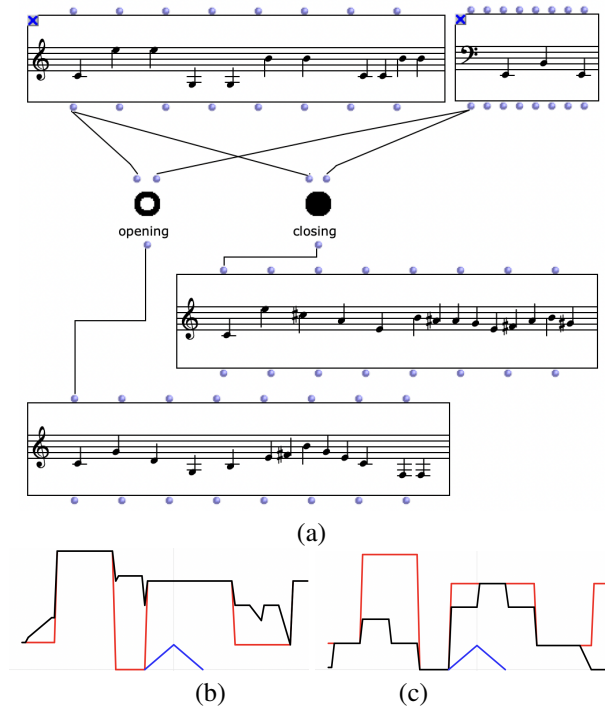


Figure 5. Opening and closing of melodies.

```
(defgeneric supremum ((obj1 t) (obj2 t)))
(defgeneric infimum ((obj1 t) (obj2 t)))
(defgeneric minus ((obj t)))
(defgeneric translate ((obj t) (x t)))
```

For instance, the binary morphology used for rhythms in Section 3 was defined by the following code:

```
(defmethod supremum ((obj1 list) (obj2 list))
  (union obj1 obj2))
(defmethod infimum ((obj1 list) (obj2 list))
  (intersect obj1 obj2))
(defmethod minus ((obj list))
  (loop for item in obj collect (* item -1)))
(defmethod translate ((obj list) (x integer))
  (loop for item in obj collect (+ item x)))
```

OMM also implements the protocol for working in grayscale morphology, that is, morphology where the lattice used is made of functions. Musically, this allows us to work with melodies if we model them as functions from time to pitches. For instance, in Figure 5 (a), we see a program that performs the opening and the closing of one melody by another one. We observe that the results produce new melodies where the range of the melodic contour has been modified depending on the structuring melody, a fifth in this example. To better understand these new melodies, let us note that in the case of the opening (Figure 5 (b)), the narrow peaks of the melodic contour were removed and the slope of jumps at the top reduced. In turn, the closing operation (Figure 5 (c)) eliminated narrow valleys and reduced the slope of jumps at the bottom. The generation of new melodies using MM operators remains highly experimental and heavily depends on the structuring melody. We have noticed a loss of control when the number of notes increases (i.e., from 4 notes onwards).

5. CONCLUSIONS

We have provided an overview of a new OpenMusic library called OMM and argued for the utility of mathematical morphology in the analysis and generation of musical structures. In this article, we have primarily highlighted the relevance of MM operators for the analysis and generation of rhythmic structures. Integrating these techniques into the OpenMusic computer-assisted composition language allows for testing the theoretical results stemming from this algebraic formalization while also providing composers or analysts with the opportunity to interact at various levels of the calculation process.

The quest for a unified method to create and transform any musical material represents, in our point of view, one of the most stimulating challenges of this musical formalization. In addition to rhythmic canons and melodies, OMM defines methods for working with lattices of rhythmic trees [11]. Details on implementations and experiments will be the subject of future work.

6. REFERENCES

- [1] C. Agón, “OpenMusic : un langage visuel pour la composition musicale assistée par ordinateur,” PhD., Paris 6, 1998.
- [2] G. Matheron, *Éléments pour une théorie des milieux poreux*. Masson et Cie, 1967.
- [3] J. Serra, *Image Analysis and Mathematical Morphology*. Academic Press, 1982.
- [4] C. Agón, M. Andreatta, J. Atif, I. Bloch, and P. Mascaraide, “Musical Descriptions Based on Formal Concept Analysis and Mathematical Morphology,” in *23rd International Conference on Conceptual Structures*. Springer, 2018, pp. 105–119.
- [5] G. Romero Garcia, “Mathematical Morphology for the Analysis and Generation of Time-Frequency Representations of Music,” PhD, Sorbonne University, 2023.
- [6] P. Lascabettes, “Mathematical Models for the Discovery of Musical Patterns, Structures and for Performances Analysis,” PhD, Sorbonne University, 2023.
- [7] I. Bloch, H. Heijmans, and C. Ronse, “Mathematical Morphology,” in *Handbook of Spatial Logics*. Dordrecht, The Netherlands: Springer, 2007, pp. 857–944.
- [8] B. A. Davey and H. A. Priestley, *Introduction to lattices and order*. Cambridge: Cambridge University Press, 1990.
- [9] D. T. Vuza, “Supplementary Sets and Regular Complementary Unending Canons,” *Perspectives of New Music*, vol. 29, 30, 31, 1991-1992.
- [10] D. Andreatta, “Constructing and Formalizing Tiling Rhythmic Canons : A Historical Survey of a ‘Mathemusal’ Problem,” *Perspectives of New Music*, vol. 1-2, n° 49, 2011.
- [11] C. Agon, K. Haddad, and G. Assayag, “Representation and rendering of rhythm structures,” in *WEDELMUSIC 2002. Proceedings.*, 2002, pp. 109–113.