Mathematical Models for the Discovery of Musical Patterns, Structures and for Performances Analysis

Paul Lascabettes

Doctoral Thesis from Sorbonne Université

École doctorale d'informatique, télécommunications et électronique (edite de paris)

Mathematical Models for the Discovery of Musical Patterns and Structures, and for Performances Analysis

---

Paul Lascabettes

Defended on November 24th, 2023

Musical Representation Team
Institut de Recherche et Coordination Acoustique Musique (IRCAM)
Sciences et Technologies de la Musique et du Son (STMS UMR 9912)

Isabelle Bloch	Professeur des universités (LIP6, Sorbonne Université)	Supervisor
Elaine Chew	Professor (KCL)	Co-supervisor
Mathieu Giraud	Directeur de recherches (CRIStAL, CNRS)	Reviewer
David Meredith	Professor (AAU)	Reviewer
Moreno Andreatta	Directeur de recherches (IRMA, CNRS)	Examiner
Gérard Assayag	Directeur de recherches (STMS, CNRS)	Examiner
Olivier Lartillot	Senior Researcher (RITMO)	Examiner
Hugues Talbot	Professeur des universités (Centrale Supelec)	Examiner

Abstract

This manuscript focuses on the application of mathematical theories to music. In particular, we explore three closely related musical components: pattern, structure and performance, which can be associated with the following questions: How to discover musical patterns? How are these patterns organized within the musical structure? And finally, how are these patterns interpreted during musical performance?

First, this manuscript deals with the musical pattern discovery task. Previous works identify two different approaches: the sequential and the multidimensional approaches. We propose a method for developing the multidimensional approach. In particular, we demonstrate that the theory of mathematical morphology fits into this context. This makes it possible to obtain mathematical results for the discovery of musical patterns, and to provide answers to some of the open questions in the field.

In the second part, we focus on the musical segmentation task, and propose to generate hierarchical segmentations. We develop a method based on homogeneity and novelty, and one based on repetition, which are the main features to be studied in order to discover the segmentation of a piece. The homogeneity-based method uses morphological filters to detect blocks on the diagonal of the self-similarity matrix, while the repetition-based method discovers almost repetitions without overlaps to obtain the hierarchical segmentations of a musical piece.

The third part is dedicated to the computational models for musical performances. We focus on the MazurkaBL dataset, which contains annotations of over 2000 recorded performances of 46 Chopin Mazurkas. To analyze this dataset, we propose to represent a musical performance in a 2-simplex, allowing us to characterize and interpret the expressivity of a performance. Then, we show that the theory of unbalanced optimal transport provides a tolerance to compare musical performances in order to identify similarities and differences between interpretations of the same piece.

Finally, we conclude this manuscript with a section presenting a scientific outreach activity on the links between mathematics and music, and a discussion of the application of mathematical theories to music.

Keywords: Musical pattern discovery, mathematical morphology, music structure, music segmentation, musical performance

Résumé

Ce manuscrit se focalise sur l'application de théories mathématiques à la musique. En particulier, nous étudions trois composantes musicales étroitement liées : les motifs, la structure et la performance, qui peuvent être associées aux problématiques suivantes. Comment découvrir les motifs musicaux ? Comment ces motifs sont-ils organisés au sein de la structure musicale ? Et enfin, comment ces motifs sont-ils interprétés lors de la performance musicale ?

La première partie est consacrée à la tâche de découverte de motifs musicaux. Les travaux antérieurs permettent de distinguer deux approches : l'approche séquentielle et l'approche multidimensionnelle. Nous proposons une méthode pour développer l'approche multidimensionnelle. En particulier, nous montrons que la théorie de la morphologie mathématique s'inscrit très bien dans ce contexte. Cela permet d'obtenir des résultats mathématiques pour découvrir des motifs musicaux, et d'apporter des réponses à certaines questions ouvertes dans ce domaine.

En seconde partie, nous nous intéressons à la tâche de segmentation musicale en proposant de générer des segmentations hiérarchiques. Nous y développons deux méthodes, l'une basée sur l'homogénéité et la nouveauté, et l'autre sur la répétition, qui sont les principales caractéristiques à étudier pour découvrir la segmentation d'une pièce. La méthode basée sur l'homogénéité utilise les filtres morphologiques pour détecter les blocs sur la diagonale de la matrice d'auto-similarité. Alors que la méthode basée sur la répétition identifie les répétitions sans intersection pour obtenir les segmentations hiérarchiques d'une pièce.

La troisième partie est dédiée à l'analyse de la performance musicale avec des outils informatiques. Nous nous focalisons sur la base de données MazurkaBL qui contient les annotations de plus de 2000 performances de 46 Mazurkas de Chopin. Pour analyser cette base de données, nous proposons de représenter une performance musicale dans un 2-simplexe, ce qui permet de caractériser et d'interpréter l'expressivité musicale d'une performance. Nous montrons, ensuite, comment la théorie du transport optimal non-équilibré permet de comparer des performances musicales afin d'identifier les similitudes et les différences entre les interprétations d'une même pièce.

Nous concluons ce manuscrit par un chapitre présentant une activité de médiation scientifique basée sur les liens entre les mathématiques et la musique, suivi d'une discussion sur l'application de théories mathématiques à la musique.

Mots clés : Découverte de motifs musicaux, morphologie mathématique, structure musicale, segmentation musicale, performance musicale

Remerciements

La première personne que j'aimerais remercier est bien évidemment Isabelle, pour l'aide qu'elle m'a apportée tout au long de ma thèse. Je tiens à souligner son ouverture d'esprit et sa bienveillance. Cela m'a permis, entre autres, d'aller jusqu'au bout de certaines idées et de garder un super souvenir de ce doctorat. Ensuite, un grand merci à Moreno ! Ce travail mathémusical n'existerait pas sans son soutien et ses encouragements à continuer dans cette voie depuis plusieurs années. J'ai aussi eu la chance de travailler à l'Ircam dans l'équipe Représentations Musicales, où j'ai beaucoup appris. Je remercie tous les membres de cette équipe et en particulier Carlos et Gérard pour leur accueil et les discussions que nous avons eues. Mes remerciements vont également à Elaine pour m'avoir fait découvrir la recherche sur la performance musicale, ainsi qu'aux membres du jury, en particulier les rapporteurs, pour leurs remarques pertinentes et leur présence lors de la soutenance de thèse.

D'un point de vue plus personnel, j'ai rencontré des personnes formidables à l'Ircam qui m'ont beaucoup inspiré et qui sont devenus des amis. Ma première pensée va aux doctorants de l'Ircam avec lesquels j'ai partagé le master ATIAM (Constance, Clément, Yann), mais également à ceux que j'ai connu pendant la thèse (Daniel, Lenny, David, Sarah, Victor, Baptiste, Nadia, Loic, Vincent, Gonzalo, Thomas, Colette, Mathilde, Giovanni).

J'aimerais également remercier mes amis en dehors du monde de la recherche. Tout d'abord, deux Simon que je connais depuis tout petit et avec qui j'ai découvert la musique. Puis deux Thomas avec lesquels j'ai fait mes études et appris à travailler. Je n'oublie pas de remercier mes amis du milieu de l'athlétisme (Adrien, Simon, Alexandre, Sylvain, Maxence, Nicolas, Rémi, Bruno, Thibaut) et bien évidemment, Serge et François pour leur soutien et leur accompagnement.

Il y a aussi le travail de certains compositeurs (Steve, Fredrik, Maynard, Thom, Tosin, Tigran, Omar, John, Kieran, Avishai, Sufjan, Stu) et réalisateurs (Gaspar, Lars, Nicolas, Hayao, Terrence, Stanley) qui est très important pour moi et que je me dois de mentionner dans cette section.

Merci également à chaque personne qui essaiera de lire et de comprendre ce qu'il y a dans ce manuscrit (courage).

Et pour finir en beauté, je remercie mes parents et mon frère sans qui rien de tout cela n'aurait été possible.

Contents

Introduction

Introduction

"Music is math" is a statement I heard a lot over the last few years. This statement always intrigued me because it is less common to hear "cinema is math" or "literature is math". This raises the question of why music is an art that can easily be associated with mathematics. There are indeed many links between mathematics and music, and they are not new. For example, in 1739, the mathematician Euler proposed the Tonnetz, a geometrical representation of musical notes as a graph [Euler 1739]. Numerous other mathematical representations and characterizations were developed in the early 2000s to provide a better understanding of certain musical concepts [Cohn 1997, Douthett 1998, Toussaint 2005, Tymoczko 2010]. Based on these previous works, this manuscript explores some of the links between mathematics and music, focusing on the application of mathematical theories to music. In particular, we explore three closely related musical components: pattern, structure and performance; which can be associated with the following questions:

Musical patterns: What are the patterns in a piece of music?

Musical structure: How are these patterns organized?

Musical performance: How are these patterns interpreted in performance?

To provide answers to these questions using mathematical and computational approaches, we use a symbolic representation of music. As opposed to the signal representations, symbolic representations of music can describe a musical score and produce data that are easily encoded and modeled, which is pertinent for mathematical applications. In particular, we use a multidimensional representation of music to discover patterns (in Part I), a string representation of music for structure (in Part II), and tempo and loudness information to analyze performance (in Part III). These different applications exemplify the *why* and *how* of employing mathematical theories to music.

Why? The application of mathematical theories to music first provides a *deeper understanding of music*, such as the organization of patterns in a piece of music or the expressivity of musical performances. Moreover, we then demonstrate the potential for *developing certain research domains*, such as the discovery of musical patterns by generalizing fundamental concepts. Finally, and perhaps most importantly, apply a mathematical approach to music leads to *mathematical results developed from a musical motivation*, like the many lemmas and theorems in this manuscript, which may also develop and enrich certain mathematical theories.

How? To illustrate how to apply mathematical theories to music, we propose *original applications of mathematical theories to music*, particularly those rarely applied to music, such as mathematical morphology (with the binary and grayscale framework) and unbalanced optimal transport (with the discrete case). In addition, we introduce various relevant *musical representations that can be modeled by mathematical objects*. These representations include onsets (the starting points of musical events), chord contours (the shape of chord sequences), adapted correlative matrix (representing non-overlapping patterns), rhythm and loudness simplices (a 2-dimensional representation of rhythm or loudness data) and boundary credence (illustrating segmentation probabilistically).

In the remainder of this introduction, we detail the three main parts of this manuscript. For each part, we introduce the problem and the motivations, situating this in relation to existing research, and present our contributions. There is also a short independent part in this manuscript, Part IV, which presents a science outreach activity on the links between mathematics and music realized during this PhD.

Part I. Musical Pattern Discovery

The first part of this manuscript focuses on the discovery of musical patterns from a multidimensional representation of music. The problematic of this part is summarized as follows.

How to discover patterns from a multidimensional representation of music?

Previous works related to this question rely on the SIA algorithm, which consists in discovering the *maximal translatable patterns* [Meredith 2002a]. Many variants of this algorithm exist and are reviewed in this part. We propose to develop some of the ideas present in these algorithms to improve the discovery of musical patterns from a multidimensional representation of music. In particular, we present a generalization of certain fundamental concepts, such as maximal translatable patterns, and provide answers to some of the open questions of this domain. The originality of our contributions results from the use of mathematical morphology, a theory mostly used in image processing with few applications to music. In particular, we demonstrate that the principal operators of binary mathematical morphology fit perfectly into the framework of musical pattern discovery, and that this mathematical theory provides relevant results for this task.

This part is divided into three chapters. Chapter 1 defines the main operators of mathematical morphology and proposes an application of this theory to various symbolic representations of music. Chapter 2 establishes the links between mathematical morphology and existing algorithms for the discovery of musical patterns. Finally, Chapter 3 proposes original mathematical results for the discovery of musical patterns using mathematical morphology.

Part II. Musical Structure

The second part of this manuscript explores the discovery of hierarchical musical segmentations, which is a particular case of musical structure. The segmentation task consists in covering the piece with a union of non-overlapping segments. We propose here to generate several segmentations of a piece in a hierarchical way, based on a symbolic representation of music. This hierarchy provides several levels of detail, with boundaries of different strengths. Therefore, the problematic of this part is stated as follows.

How to discover hierarchical segmentations from symbolic representations?

Previous works on the discovery of musical segmentation have identified three major components for this task: *repetition*, *homogeneity* and *novelty* [Paulus 2010]. The repetition criterion is based on the fact that the repetition of musical elements has a strong impact on the segmentation of a piece. In particular, Cambouropoulos asserted that "the beginning and ending points of significant repeating musical patterns influence the segmentation of a musical surface" [Cambouropoulos 2006]. The homogeneity criterion is based on the fact that a change in the nature of the musical elements has an impact on the segmentation of a piece. For example, if the instruments change at a given moment, this gives us information about a segmentation boundary. Finally, the novelty criterion is based on the fact that a new element produces information on a segmentation boundary. The homogeneity criterion is therefore closely related to the novelty criterion. In this part, we propose two methods, the first is based on homogeneity and novelty, and the second on repetition. In this way, this part fits into the context of previous works. However, since most of the works on segmentation discovery are based on audio, our approach is original because it uses a symbolic representation of music.

This part is divided into two chapters. Chapter 4 demonstrates the relevance of grayscale mathematical morphology for the homogenization step, where a small morphological filter homogenizes small sections, while a large filter homogenizes the global parts of a musical piece. Chapter 5 generalizes the correlative matrix to discover non overlapping repeated patterns and propose to extract the information contained in this matrix to generate hierarchical segmentations.

Part III. Musical Performance

The third part of this manuscript focuses on the analysis of musical performance using computational models. Cancino-Chacón et al. [Cancino-Chacón 2018] describe three main components of the performance process: *interpretation* (the act of understanding the structure, such as grouping and segmentation of sequences); *planning* (the performer decides how to communicate the musical structure, for example, through the use of tempo archs); *movement* (physical movement which has an impact on the way humans perform and perceive music). In this part, we focus on the

first two components: interpretation and planning. In addition, we choose to design models for solo piano performances. This choice is due to the databases, which are mostly composed of solo piano performances, and to the simplicity of modeling, because the analysis can be restricted to tempo and loudness variations without losing too much information. Therefore, the problematic of this part is formulated as follows.

How to analyze musical performance using tempo and loudness information?

To do this, we focus on the MazurkaBL dataset, which is currently the largest database of solo piano performances [Kosta 2018]. It contains beat-level tempo and loudness data from over 2000 performances of 46 Mazurkas, which is highly appropriate for our purposes of comparison and performance analysis.

This part is divided into two chapters. Chapter 6 proposes an original visualization of the expressivity of musical performance using the rhythm or loudness simplex. Chapter 7 presents a method for discovering and comparing segmentations induced by musical performance, based on tempo or loudness variations.

List of Publications

[Lascabettes 2024] **Paul Lascabettes** & Isabelle Bloch. *Discovering Repeated Patterns From the Onsets in a Multidimensional Representation of Music*, International Conference on Discrete Geometry and Mathematical Morphology (DGMM 2024).

[Guichaoua 2024] Corentin Guichaoua, **Paul Lascabettes** & Elaine Chew. *End-To-End Bayesian Segmentation and Similarity Assessment of Performed Music Tempo and Dynamics With No Score Information*, Music & Science, 2024.

[Lascabettes 2023] **Paul Lascabettes**, Elaine Chew & Isabelle Bloch. *Characterizing and Interpreting Music Expressivity through Rhythm and Loudness Simplices*, International Computer Music Conference (ICMC 2023).

[Lascabettes 2022b] **Paul Lascabettes**, Corentin Guichaoua & Elaine Chew. *Generating Multiple Hierarchical Segmentations of Music Sequences Using Correlative Matrices*, Sound and Music Computing Conference (SMC 2022).

[Lascabettes 2022a]	**Paul Lascabettes**, Carlos Agon, Moreno Andreatta & Isabelle Bloch. *Computational Analysis of Musical Structures based on Morphological Filters*, International Conference on Mathematics and Computation in Music (MCM 2022).
[Romero-García 2022b]	Gonzalo Romero-García, **Paul Lascabettes** & Elaine Chew. *Automated Musical Rhythm Transcription of ECG RR Interval Time Series as a Tool for Representing Rhythm Variations and Annotation Anomalies in Arrhythmia Heartbeat Classifications*, Computing in Cardiology (CinC 2022).
[Lascabettes 2020]	**Paul Lascabettes**, Isabelle Bloch & Carlos Agon. *Analyse de représentations spatiales de la musique par des opérateurs simples de morphologie mathématique*, Journées d'Informatique Musicale (JIM 2020).

Publications Under Review or in Preparation

Paul Lascabettes & Isabelle Bloch. *Links Between Point-Set Algorithms in Music and Mathematical Morphology*, Journal of New Music Research, 2024.

Paul Lascabettes & Isabelle Bloch. *Using Mathematical Morphology to Discover Repeated Patterns in Music*, Journal of Mathematics and Music, 2024.

Awards and Grants

Research on mathematical morphology applied to music was awarded the AFIM young researcher prize at the Journée d'Informatique Musicale [Lascabettes 2020]. This PhD was also awarded a grant by the French Ministry of Culture for the call for projects "Research in Music: Collaborations between Young Researchers and Artists" to develop the project "Articulation between Theoretical Research and Algorithmic Composition".

Context

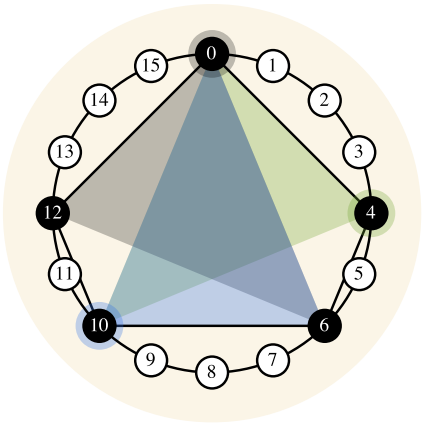
This PhD was carried out at the Institute for Research and Coordination in Acoustics/Music (IRCAM) in the Musical Representation team (RepMus) as part of the Science and Technology of Music and Sound (STMS) joint research unit (UMR9912). This research was funded by a specific doctoral contract for normaliens (CDSN) via Sorbonne University (SU).

Part I

Musical Pattern Discovery

Chapter 1

Mathematical Morphology Applied to Symbolic Representations of Music



Contents

In this chapter, we introduce binary mathematical morphology, a theory mainly applied to image processing. We present the principal operators: dilation, erosion, opening and closing. For each operator, we list the properties that are useful in

the next chapters and provide an interpretation of the results applied to an image. The originality of this chapter resides in the application of this theory to symbolic representations of music. We propose various applications to different musical data: chords, rhythms and musical pieces. This is a new and original approach to an algebraic formalism applied to music.

Section 1.1 presents the context and related works of mathematical morphology applied to music. Section 1.2 introduces the main definitions and properties of binary mathematical morphology theory. Section 1.3 proposes to apply mathematical morphology to symbolic representations of music, where the relevance of each morphological operator is reviewed.

Some of the ideas presented in this chapter are from the article "Analyse de représentations spatiales de la musique par des opérateurs simples de morphologie mathématique" [Lascabettes 2020].

1.1 Context for Mathematical Morphology Applied to Music

Developed in the 1960s by Matheron and Serra at the École des Mines de Paris, mathematical morphology has become a research area of growing interest in the international scientific community [Matheron 2002]. This is an algebraic theory that analyzes shapes and is mostly used in image analysis and understanding. However, this theory is not very common yet in the Mathematics and Music community. The fundamental idea of this theory is to modify the shape, the size or the topological properties of objects with non-linear and non-reversible transformations. Among the few existing applications of mathematical morphology to symbolic representations of music, automatic methods have been developed to detect approximate occurrences of musical patterns in symbolic musical databases [Karvonen 2008, Karvonen 2010]. In this case, mathematical morphology enables to match almost identical patterns. Moreover, mathematical morphology has been used to analyze concept lattices based on musical intervals [Agon 2018], and to obtain the musical structure by filter self-similarity matrices with morphological operators [Lascabettes 2022a]. Finally, morphology has also been applied to analyze and generate musical data. In this context, principal operators of mathematical morphology have been adapted to find a musical meaning, allowing for example extracting harmonic components or to obtain musical transformations [Lascabettes 2019, Lascabettes 2020, Romero-García 2022a]. This chapter is a continuation of this work, with the aim of better understanding how to apply the principal operators to symbolic representations of music.

1.2 Binary Mathematical Morphology

In this section, we briefly recall the concepts of mathematical morphology that are useful in the following chapters. In particular, only the fundamental notions

of binary mathematical morphology are introduced here; further information in a more general setting can be found in [Heijmans 1990, Ronse 1991, Heijmans 1994, Bloch 2007, Najman 2010].

1.2.1 Dilation and Erosion

1.2.1.1 Definitions

Most deterministic morphological transformations derive from two basic operations: *dilation* and *erosion*. In general, in a complete lattice, a dilation is defined as an operator that commutes with the supremum, and an erosion as an operator that commutes with the infimum (in the sense of the partial order defined on the lattice). These abstract definitions are often specified in a concrete form involving a structuring element, representing a binary relationship. We consider here the simple case of the lattice of parts of a set E, where E is $\mathbb{R}^n$, $\mathbb{Z}^n$ or $\mathbb{Z}_n$ provided with the inclusion relation, i.e. $(\mathcal{P}(E), \subseteq)$, and the usual addition of E. First of all, some basic notions are required. Let $X \in \mathcal{P}(E)$ (i.e. $X \subseteq E$), recall that:

- The *complement* of X is $X^c = \{x \in E \mid x \notin X\}$.
- The *translation* of X by $t \in E$ is $X_t = \{x + t \mid x \in X\}$.
- The *symmetrical* of X is $\check{X} = \{-x \mid x \in X\}$.

This allows us to define the principal operators of mathematical morphology.

Definition 1.1: Dilation and Erosion

Let $S \in \mathcal{P}(E)$, the **dilation** δ_S and **erosion** ε_S by S are defined by:

$$\begin{array}{rccll} \delta_S: & \mathcal{P}(E) & \longrightarrow & \mathcal{P}(E) & \\ & X & \longmapsto & X \oplus S & = \{x + s \mid x \in X, s \in S\} \\ & & & & = \{x \in E \mid \check{S}_x \cap X \neq \emptyset\} \\ \varepsilon_S: & \mathcal{P}(E) & \longrightarrow & \mathcal{P}(E) & \\ & X & \longmapsto & X \ominus S & = \{x \in E \mid S_x \subseteq X\} \end{array}$$

Where $\oplus$ and $\ominus$ are respectively the *Minkowski addition* [Minkowski 1903] and *subtraction* [Hadwiger 1950]. In addition, S is called a *structuring element*. Note that the structuring element is defined with an *origin* O_E which has a direct impact on Minkowski addition and subtraction, and therefore modifies the result of dilation and erosion.

1.2.1.2 Properties

Dilation and erosion satisfy several mathematical properties, but it is first important to note that these operations are non-reversible and non-linear. We list the fundamental properties that are useful in the following, but other properties can be found in the cited literature. In all the properties stated below, it is assumed that $X, X', S, S' \in \mathcal{P}(E)$.

- These two operations are *dual by complementation*:

$$X \oplus S = (X^c \ominus \check{S})^c, \qquad X \ominus S = (X^c \oplus \check{S})^c$$

- If and only if the origin is included in the structuring element, dilation is *extensive* while erosion is *anti-extensive*:

$$O_E \in S \iff X \subseteq \delta_S(X), \qquad O_E \in S \iff \varepsilon_S(X) \subseteq X$$

- An important property is that these operations are *increasing*:

$$X \subseteq X' \Rightarrow \delta_S(X) \subseteq \delta_S(X'), \qquad X \subseteq X' \Rightarrow \varepsilon_S(X) \subseteq \varepsilon_S(X')$$

- The dilation is *increasing according to the structuring element* while the erosion is *decreasing according to the structuring element*:

$$S \subseteq S' \Rightarrow \delta_S(X) \subseteq \delta_{S'}(X), \qquad S \subseteq S' \Rightarrow \varepsilon_{S'}(X) \subseteq \varepsilon_S(X)$$

- These two operations also verify the *adjunction* property:

$$\delta_S(X) \subseteq X' \Leftrightarrow X \subseteq \varepsilon_S(X')$$

- It is easy to check that the dilation is *commutative*:

$$\delta_S(S') = \delta_{S'}(S)$$

- Finally, these operations satisfies also the *iteration property*:

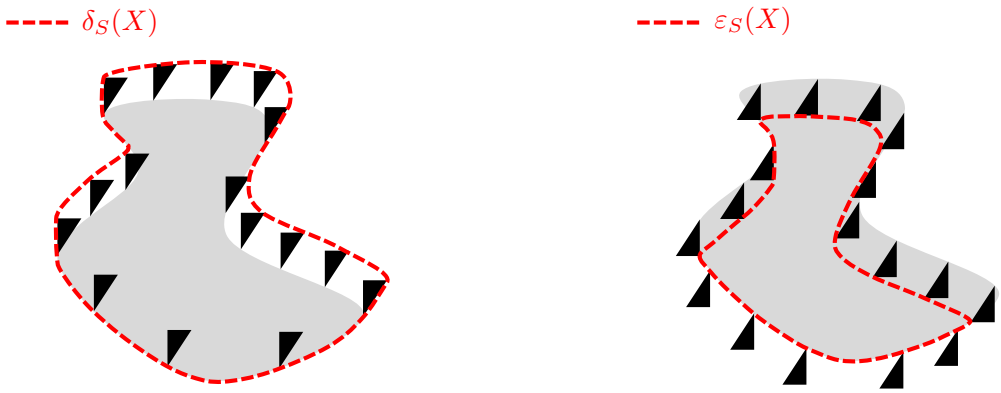
$$\delta_S(\delta_{S'}(X)) = \delta_{S \oplus S'}(X), \qquad \varepsilon_S(\varepsilon_{S'}(X)) = \varepsilon_{S \oplus S'}(X)$$

1.2.1.3 Interpretations

To illustrate these concepts, a set X and a structuring element S are illustrated in Figure 1.1. By default, the origin may be located in the center of the structuring element, but in our example, the origin is located in the lower corner of the triangle. To understand the shape obtained after a dilation, consider that the origin of the structuring element moves along X. In this case, the dilation is composed of all the

(a) X (b) Structuring element S

Figure 1.1: Example of a set X in light grey (left) and a structuring element S (right) represented with its origin.



(a) Dilation (b) Erosion

Figure 1.2: Dilation and erosion of X by S (as defined in Figure 1.1) represented by dotted red lines.

points covered by the structuring element. Figure 1.2(a) illustrates the transformation produced by a dilation where the boundary of the dilated set is displayed by a dotted red line. Regarding the morphological erosion, there are two approaches to interpret the result. First, this can be interpreted as the set of points in X where the structuring element is included in X when its origin is placed on these points. It is also possible to use the property of duality by complementation with the dilation to interpret the erosion, i.e. consider an erosion as a dilation of the complementary of X by the symmetrical of the structuring element. The origin of the structuring element moves along the boundary of X, but the structuring element is reversed and the boundary of the eroded set is the set of points that the structuring element

cannot cover. The boundary of the erosion is displayed by a dotted red line in Figure 1.2(b). In both cases, the shape of X has changed and the transformations are not just homotheties. Therefore, a dilation fills holes smaller than the structuring element (in the sense of inclusion), welds components that are topologically close and expands shapes. By contrast, erosion eliminates connected components smaller than the structuring element, enlarges holes and reduces object size.

1.2.2 Opening and Closing

1.2.2.1 Definitions

The other two fundamental operations result from the composition of the two previous ones. The *opening* γ_S is the composition of an erosion and a dilation, and the *closing* φ_S is a dilation followed by an erosion.

Definition 1.2: Opening and Closing

Let $S \in \mathcal{P}(E)$, the **opening** γ_S and the **closing** φ_S by S are defined by:

$$\begin{array}{rccl} \gamma_S : & \mathcal{P}(E) & \longrightarrow & \mathcal{P}(E) \\ & X & \longmapsto & X \circ S = (X \ominus S) \oplus S \\ \\ \varphi_S : & \mathcal{P}(E) & \longrightarrow & \mathcal{P}(E) \\ & X & \longmapsto & X \bullet S = (X \oplus S) \ominus S \end{array}$$

The opening can be described simply [Serra 1982]. Let $X, S \in \mathcal{P}(E)$, we have:

$$\gamma_S(X) = \bigcup \{S_x \mid x \in E \wedge S_x \subseteq X\}$$

Note that the origin of the structuring element is no longer important for these two operations, only the shape is involved.

1.2.2.2 Properties

As with dilation and erosion, we list some important mathematical properties satisfied by the opening and closing, with $X, X', S, S' \in \mathcal{P}(E)$. As before, it is important to specify that these operations are non-linear and non-reversible.

- These operations are *dual by complementation*:

$$X \circ S = (X^c \bullet \check{S})^c, \qquad X \bullet S = (X^c \circ \check{S})^c$$

- One of the most important properties of the opening and closing operators is that they are both *idempotent*:

$$\gamma_S \circ \gamma_S(X) = \gamma_S(X), \qquad \varphi_S \circ \varphi_S(X) = \varphi_S(X)$$

- Unlike dilation and erosion, which require the origin to be included in the structuring element, in all cases the opening is *anti-extensive* and the closing is *extensive*:

$$\gamma_S(X) \subseteq X, \qquad X \subseteq \varphi_S(X)$$

- The opening is *decreasing according to the structuring element* while the closing is *increasing according to the structuring element*:

$$S \subseteq S' \Rightarrow \gamma_{S'}(X) \subseteq \gamma_S(X), \qquad S \subseteq S' \Rightarrow \varphi_S(X) \subseteq \varphi_{S'}(X)$$

- These two operations are *increasing* (as compositions of increasing operations):

$$X \subseteq X' \Rightarrow \gamma_S(X) \subseteq \gamma_S(X'), \qquad X \subseteq X' \Rightarrow \varphi_S(X) \subseteq \varphi_S(X')$$

Table 1.1 summarizes the main properties that satisfy each of the four principal operations of mathematical morphology (dilation, erosion, opening, closing).

Table 1.1: Summary of the main properties of the four principal operators of mathematical morphology, where ✓* assumes that the origin is included in the structuring element.

Properties	δ	ε	φ	γ
Dual by complementation	✓	✓	✓	✓
Extensive	✓*		✓	
Anti-extensive		✓*		✓
Increasing	✓	✓	✓	✓
Increasing according to S	✓		✓	
Decreasing according to S		✓		✓
Iteration	✓	✓		
Idempotent			✓	✓
Commutative	✓			
Analysis operator		✓		✓
Generative operator	✓		✓	

In addition to the basic properties just described, we characterize erosion and opening as *analysis operators*, in the sense that they are anti-extensive operators (or most of the time for erosion) and are therefore relevant to musical analysis. In contrast, we characterize dilation and closing as *generative operators*, in the sense that they are extensive operators (or most of the time for dilation), i.e. they generate musical data from the dataset and the structuring element.

By successively composing opening and closing with the same structuring element, we can obtain four new filters: opening followed by closing, closing followed by opening, opening followed by closing and opening, closing followed by opening

and closing. These four operations are idempotent, and no other operator can be obtained by additional composition [Serra 1988]. Finally, Figure 1.3 illustrates some of the links between the four principal operations of mathematical morphology.

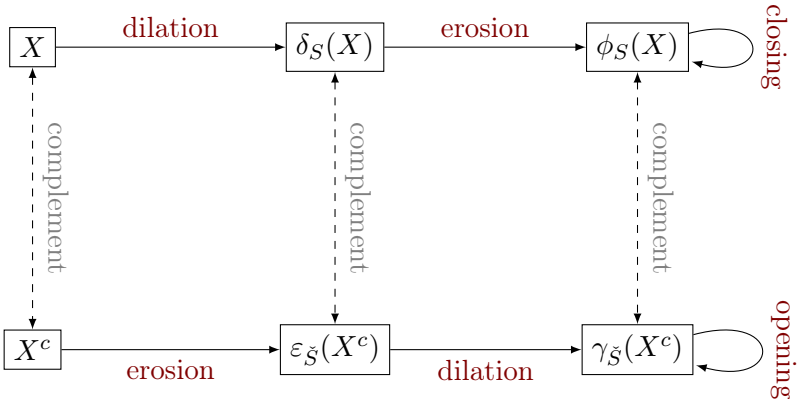
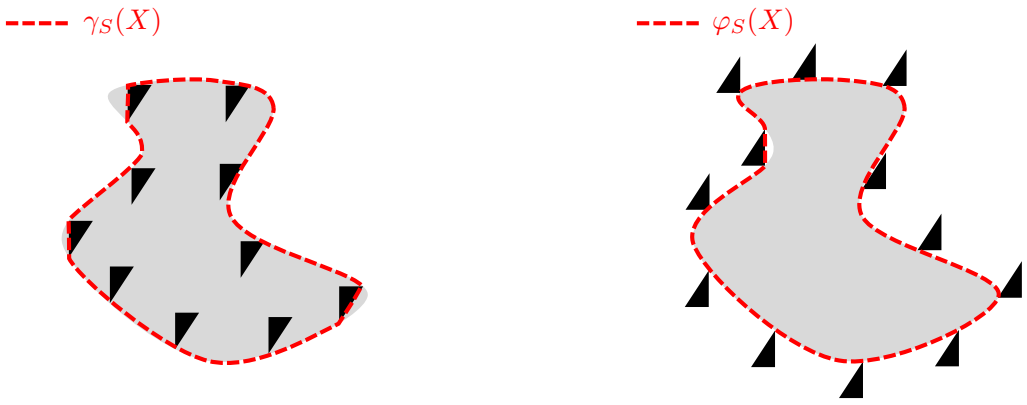


Figure 1.3: Illustrations of the links between the four principal operators of mathematical morphology.

1.2.2.3 Interpretations

The opening of X can be interpreted as the set of points covered by the structuring element when it is included in X. By duality, the closing of X consists in adding points to X where the symmetrical of the structuring element cannot cover by remaining in the complementary of X. The boundaries of the results of these operations are represented by dotted red lines in Figure 1.4 using the same set X and structuring element S as in Figure 1.1. An opening regularizes shapes, removes



(a) Opening (b) Closing

Figure 1.4: Opening and closing of X by S (as defined in Figure 1.1) represented by dotted red lines.

connected components smaller than the structuring element, does not (always) preserve topology and reduces the object. Whereas a closing fills holes smaller than the structuring element, does not (always) preserve topology, welds shapes together and enlarges the object.

To summarize, here is the interpretation of the four basic operations of the mathematical morphology of the set X by the structuring element S.

- **Dilation:** Union of all translations of S to positions in X.
- **Erosion:** All positions at which S occur in X.
- **Opening:** Union of all occurences of S in X.
- **Closing:** All positions at which S occurs in the union of all translations of S to positions in X.

1.3 Applications of the Principal Morphological Operators to Symbolic Representations of Music

In this section, we propose to apply the principal morphological operators to symbolic representations of music. We demonstrate the possibility of interpreting morphological operators musically. To do this, the set X represents the musical data, while the structuring element is a musical property that is applied to X, where the different operations yield different results. These simple examples, initially described in [Lascabettes 2019, Lascabettes 2020, Romero-García 2022a], introduce an original approach, combining mathematical (essentially algebraic) formulations with musical data. We demonstrate that morphological operators can be applied to several types of musical data covering different representations, from pitch sets ($E = \mathbb{Z}$), musical rhythms ($E = \mathbb{Z}_n$) or musical scores ($E = \mathbb{R}^n$).

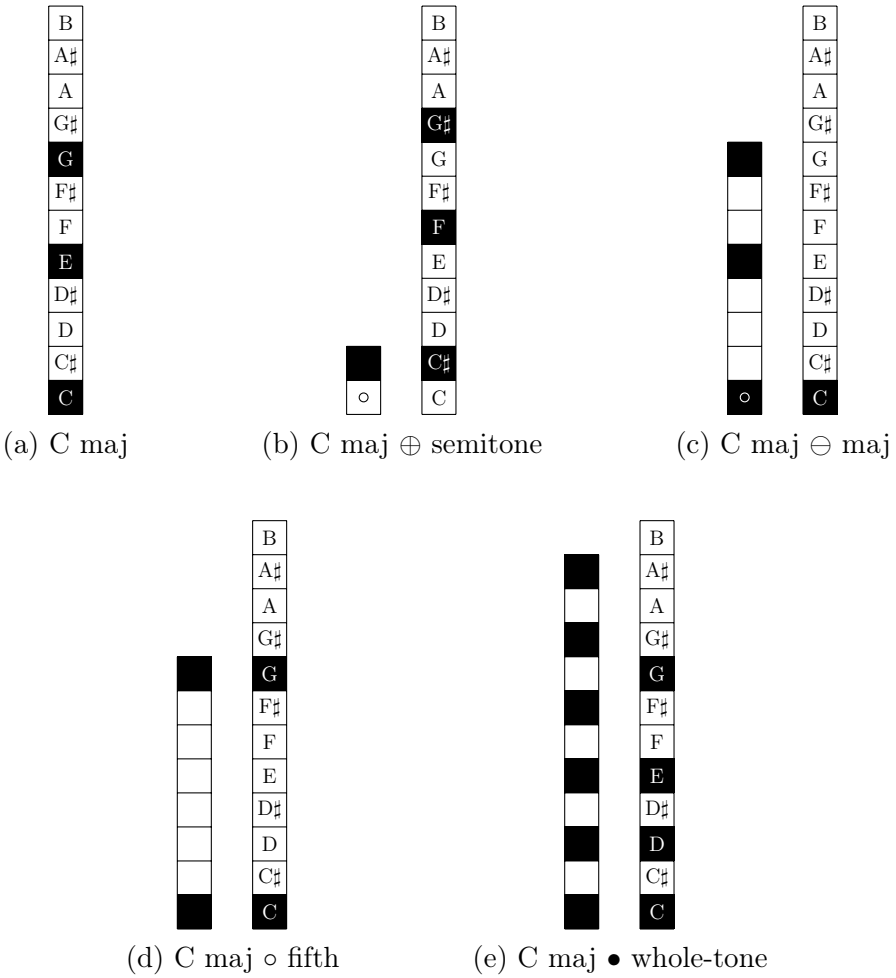
1.3.1 Application of the Principal Morphological Operators to a Major Chord

Before applying mathematical morphology to musical pieces, we need to understand the transformations that can be achieved with a chord. The temporal dimension is not considered, which reduces complexity but also potential applications of mathematical morphology. However, it is possible to obtain some results from the four principal operations of binary morphology. In this case, we consider the set of pitches with $E = \mathbb{Z}$, where subsets of E are sets of pitches that can be interpreted as chords. For example, we define C maj $= \{C_4, E_4, G_4\}$, where C_4, E_4, G_4 are respectively the pitches corresponding to C, E, G of the fourth octave. To match each pitch to an element of $\mathbb{Z}$, we arbitrarily choose C_0 as the origin, $C_0^\sharp$ as 1 and so on, such that each element a semitone away is one away in $\mathbb{Z}$. This defines each chord as a subset of $E = \mathbb{Z}$. Therefore, the aim is to use morphological operators to obtain a musical

result from a chord and a musical property, which is the structuring element. For this, we define the following structuring elements:

$$\text{maj} = \{0, 4, 7\}, \quad \text{min} = \{0, 3, 7\}, \quad \text{fifth} = \{0, 7\}, \quad \text{semitone} = \{1\}$$

In the remainder of this section, we present the relevance of each of the four principal operations of mathematical morphology applied to the C maj chord. The results are illustrated in Figure 1.5 where the pitch is represented vertically to refer to a piano roll representation. In this representation, the vertical axis refers to the pitch of the notes, while the horizontal axis indicates time, which is not used here as the elements are pitch sets. Finally, these results are not limited to the C maj chord and can be generalized to other pitch sets.



(a) C maj (b) C maj ⊕ semitone (c) C maj ⊖ maj

(d) C maj ∘ fifth (e) C maj • whole-tone

Figure 1.5: Principal morphological operators applied to a major chord (a), where the structuring element is on the left and the result on the right. The origin of the structuring element for dilation and erosion is indicated by a circle.

- **Dilation** enriches or creates a chord. For example, it is possible to obtain C maj from C_4 and the maj property with dilation:

$$\{C_4\} \oplus \text{maj} = \text{C maj}$$

In the same way, we can obtain other chords with the other structuring elements:

$$\{C_4\} \oplus \text{min} = \text{C min}, \qquad \{C_4\} \oplus \text{fifth} = \text{C5}$$

Where C min $= \{C_4, E_4^\flat, G_4\}$ and C5 $= \{C_4, G_4\}$. In addition, it is also possible to combine several structuring elements, where the order is not important because dilation is commutative. For example, if the structuring element contains a single element which is different from the origin, the dilation is equivalent to a translation, i.e. in musical terms to a transposition. Combining the previous equation with the semitone structuring element, we obtain additionally a transposition of a semitone:

$$\{C_4\} \oplus \text{maj} \oplus \text{semitone} = \text{C}\sharp \text{ maj}$$

As a general rule, the number of semitones in the transposition is equal to the difference between the position of the structuring element and its origin.

- **Erosion** extracts specific characteristics from the chord. For example, with the structuring elements maj and min, erosion extracts only the fundamental of major and minor chords. Consequently, the erosion of C maj by maj is the fundamental C_4, whereas the erosion by min is the empty set, as C maj is not composed of minor chords:

$$\text{C maj} \ominus \text{maj} = \{C_4\}, \qquad \text{C maj} \ominus \text{min} = \emptyset$$

More generally, by selecting intervals as structuring elements, erosion extracts a single picth in relation to these intervals. This pitch corresponds to the position of the origin in the structuring element. In the previous examples, erosion extracts the fundamental, but it is possible to adapt the position of the origin to extract other characteristics.

- **Opening** extracts musical intervals and simplifies the chord. For example, the opening of C maj by maj has no effect, but with fifth as the structuring element, the opening extracts only the fifth intervals from the initial chord, thus simplifying it by removing the third E_4:

$$\text{C maj} \circ \text{maj} = \text{C maj}, \qquad \text{C maj} \circ \text{fifth} = \text{C5}$$

As a general rule, only patterns similar to the structuring element are parts of the opening result. By applying this operation to a musical piece with the fifth as structuring element, it is possible to extract all the fifth intervals of the piece, which can be relevant for simplifying musical data.

- **Closing** enriches the initial chord because this operation is extensive, but this is the least intuitive morphological operation. For example, with the structuring element whole-tone $= \{0, 2, 4, 6, 8, 10\}$, which corresponds to the whole-tone scale, closing adds the pitch D_4 to the C maj chord:

$$\text{C maj} \bullet \text{whole-tone} = \{C_4, D_4, G_4, E_4\}$$

  This is because there is exactly one tone away between D_4 and C_4 and between D_4 and G_4. However, it is not possible to add just any pitch to the chord, which makes this operation more complicated to use. Indeed, in the example of a C maj chord, it is not possible to add a B_4 to form a C maj7 chord, which is very common. Nevertheless, it is possible to obtain this result with other morphological operations (in particular with the *hit-or-miss* operation [Serra 1982]).

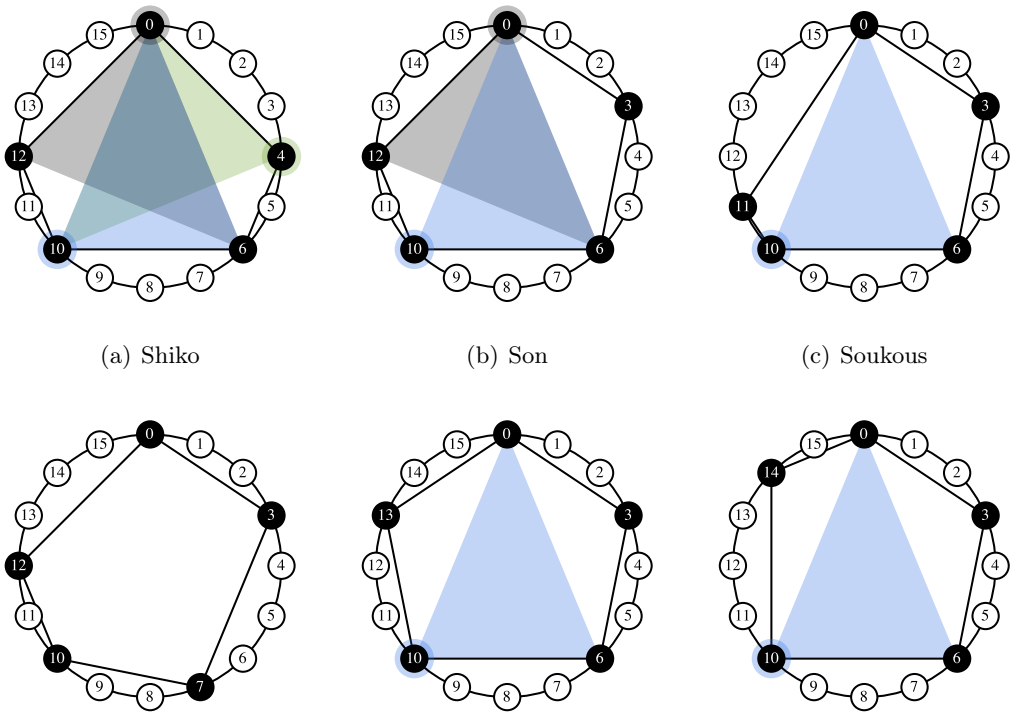
1.3.2 Application of the Principal Morphological Operators to Rhythms

In this section, we demonstrate that morphological operators can also be applied to musical rhythms. To do this, we use another data representation: the circular representation with $E = \mathbb{Z}_n$. In this case, each subset of E is a musical rhythm. We propose here to analyze the six distinguished rhythms considered by Toussaint [Toussaint 2019]. These six rhythms have been selected because they are considered as the most significant rhythms in the music of the world among the rhythms with 5 note onsets and 16 pulses. Because all these rhythms contain 16 pulses, the morphological operators are applied in $E = \mathbb{Z}_{16}$. They are composed of: Shiko $= \{\overline{0}, \overline{4}, \overline{6}, \overline{10}, \overline{12}\}$, Son $= \{\overline{0}, \overline{3}, \overline{6}, \overline{10}, \overline{12}\}$, Soukous $= \{\overline{0}, \overline{3}, \overline{6}, \overline{10}, \overline{11}\}$, Rumba $= \{\overline{0}, \overline{3}, \overline{7}, \overline{10}, \overline{12}\}$, Bossa-Nova $= \{\overline{0}, \overline{3}, \overline{6}, \overline{10}, \overline{13}\}$ and Gahu $= \{\overline{0}, \overline{3}, \overline{6}, \overline{10}, \overline{14}\}$. The notation $\overline{x}$ indicates that $x \in \mathbb{Z}$ is a representative of the equivalence class $\overline{x} \in \mathbb{Z}_{16}$. These rhythms are all illustrated in Figure 1.6 using a circular representation. To analyze these rhythms, we use morphological analysis operators, i.e. erosion and opening. The other two operators, dilation and closing, enrich the rhythm, which is not the purpose here.

We chose the Tresillo $= \{\overline{0}, \overline{6}, \overline{12}\} \in \mathcal{P}(E)$ as a structuring element, with the origin naturally positioned on the $\overline{0}$. This is a very common rhythmic pattern and it may be interesting to explore its relationship with the six distinguished rhythms. Morphological operators reveal the inclusion relationships between the Tresillo and the six distinguished rhythms. For example, the erosion and opening of the Shiko by the Tresillo provide:

$$\text{Shiko} \ominus \text{Tresillo} = \text{Tresillo} + \overline{4}, \qquad \text{Shiko} \circ \text{Tresillo} = \text{Shiko}$$

Where Tresillo $+ \overline{4} = \{\overline{0}, \overline{4}, \overline{10}\}$ is a Tresillo shifted from $\overline{4}$. In this way, the opening of the Shiko with the Tresillo makes it clear that the Shiko is a rhythm composed of Tresillos. Moreover, by examining with erosion when the Tresillo appear in the



(a) Shiko (b) Son (c) Soukous

(d) Rumba (e) Bossa-Nova (f) Gahu

Figure 1.6: The six distinguished rhythms with 5 note onsets and 16 pulses analyzed by Toussaint [Toussaint 2019]. Colored polygons indicate the inclusions of Tresillos in these rhythms, which is useful for understanding morphological results.

Shiko, we surprisingly also get a Tresillo but time-shifted. These results are illustrated in Figure 1.6(a), where the Tresillos inclusions in the Shiko are visualized by three colored polygons. The positions $\overline{0}, \overline{4}$ and $\overline{10}$ are highlighted because they are the origins of the three Tresillos included in the Shiko, which is the result of the erosion of the Shiko by the Tresillo.

Next, Rumba is the only rhythm of the six distinguished rhythms that does not contain the Tresillo:

$$\text{Rumba} \circ \text{Tresillo} = \emptyset$$

In addition, there is only one inclusion of Tresillo in Soukous, Bossa-Nova and Gahu, and it is in the same position:

$$\text{Soukous} \ominus \text{Tresillo} = \text{Bossa-Nova} \ominus \text{Tresillo} = \text{Gahu} \ominus \text{Tresillo}$$

Finally, with the exception of Rumba, all the rhythms of the six distinguished rhythms contain a Tresillo that starts at $\overline{10}$. Formally, $\forall$ Rhythm $\in$ {Shiko, Son, Soukous, Bossa-Nova, Gahu}:

$$\overline{10} \in \text{Rhythm} \ominus \text{Tresillo}$$

This is indicated by the light blue triangle included in the rhythms in Figure 1.6. These simple examples illustrate how morphological operators can be used to analyze rhythmic data from the cyclic group $E = \mathbb{Z}_n$, but other morphological operators may be used to modify and generate rhythms.

1.3.3 Analysing Musical Data Using Mathematical Morphology

After proposing to apply the principal morphological operators to a chord in Section 1.3.1 (with $E = \mathbb{Z}$) and to a rhythm in Section 1.3.2 (with $E = \mathbb{Z}_n$), we focus on the analysis of a musical piece using mathematical morphology operators. In this situation, the set $E = \mathbb{R}^n$ represents the set of musical notes, where each characteristic of the musical notes is a component of the space. Subsets of E are sets of notes that can be interpreted as musical pieces. We have chosen here to restrict ourselves to the case where a note is represented by three elements: (note onset, pitch, duration), but the results presented here can be generalized to other situations. Consequently, a musical piece is represented here by a subset of $E = \mathbb{R}^3$. Since erosion and opening are analysis operators (unlike dilation and closure, which are generative operators), it is these two operations that are useful for analyzing a piece of music. To illustrate the results these operations can produce, the introduction of *Hey Jude* by The Beatles is analyzed. In order to simplify understanding, this is not the official version, but a piano adaptation, which is shown in Figure 1.7 using a piano roll representation (to avoid representing data in $\mathbb{R}^3$).

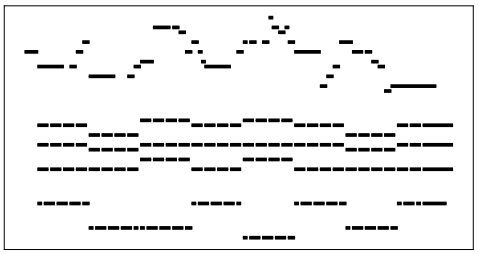


Figure 1.7: Example of a set X illustrated using a piano roll representation, where X is an adapted version of the introduction of *Hey Jude* (The Beatles).

The melody is present at the top of the piano roll because of its high notes, while the bass is located below. The chords played in this example are in the middle part of the piano roll. This example is based on major chords, so it is natural to choose major chords as structuring elements in order to extract the chords. As a major chord is composed of three notes: the root, the major third and the fifth, it can be represented in three different ways if the order is maintained, as illustrated in Figure 1.8.

By setting the origin on the root of the chord, we define the three structuring elements maj_1, maj_2 and maj_3 illustrated in Figure 1.8, which represent the three possibilities of obtaining a major chord, with:

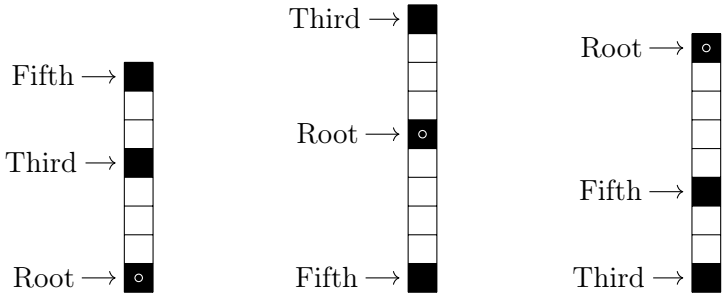


Figure 1.8: Three different representations of a major chord in the piano roll, considered as three structuring elements. The origin is chosen on the root.

- $\mathrm{maj}_1 = ((0,0,0),(0,4,0),(0,7,0))$ which corresponds to the permutation root/third/fifth, where the fundamental is the lowest note,
- $\mathrm{maj}_2 = ((0,-5,0),(0,0,0),(0,4,0))$ which corresponds to fifth/root/third,
- $\mathrm{maj}_3 = ((0,-8,0),(0,-5,0),(0,0,0))$ which corresponds to third/fifth/root.

The 0 value on the first and last coordinates does not indicate that the discovered notes are those where the note onset is 0 and the duration is 0. In particular, an opening or erosion by these structuring elements handles notes played at the same time (the first coordinate does not change) and of the same duration (the last coordinate does not change). Nevertheless, it is possible to have structuring elements where the chord is spread out over time, like arpeggios or with different durations.

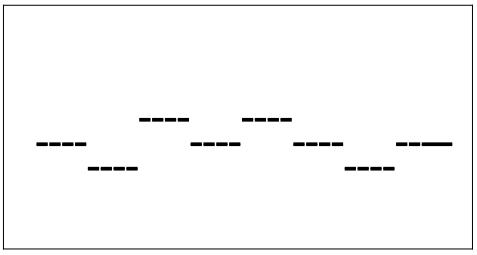
Certain morphological concepts can be generalized to several structuring elements:

$$\varepsilon_{S_1,\ldots,S_n}(X) = \bigcup_{i\in\{1,..,n\}} \varepsilon_{S_i}(X), \qquad \gamma_{S_1,\ldots,S_n}(X) = \bigcup_{i\in\{1,..,n\}} \gamma_{S_i}(X)$$

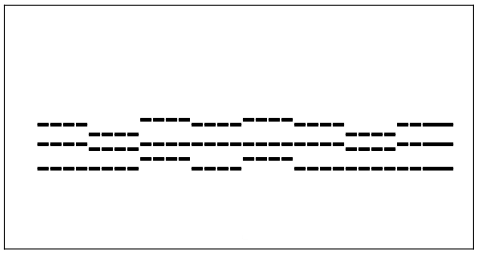
Note that a union of openings is also an opening in an algebraic sense, i.e. an increasing, anti-extensive and idempotent operation. This is not the case for the erosion but that does not prevent us from using the proposed generalization.

As described in Section 1.3.1, erosion extracts only the root of each chord, because the origin is chosen on the root. The result of erosion by the three structuring elements maj_1, maj_2 and maj_3 is illustrated in Figure 1.9(a). This reveals that the introduction to this adaptation of *Hey Jude* is composed of the following major chords: F / C / B♭ / F / B♭ / F / C / F, each of which is repeated four times.

Then, the opening by the three structuring elements maj_1, maj_2 and maj_3 extracts the major chords of the piece. The result of this operation is displayed in Figure 1.9(b), where the melody and bass have been removed after this transformation.



(a) Erosion of musical data by major chords to extract the root of the chords



(b) Opening of musical data by major chords to extract the chords

Figure 1.9: Erosion and opening applied to the introduction of *Hey Jude* by major chords as structuring elements to extract musical information.

These two examples illustrate the potential of morphological tools for extracting information and analyzing a piece of music. To extend the possibilities, it is also possible to combine morphological operations to obtain other results. For example, it is possible to remove the chords from the musical piece in order to extract the melody or bass. This can be achieved with the morphological operation *white top-hat*, which consists in subtracting the opening from the musical piece.

1.4 Conclusion and Future Work

In this chapter, we first introduced the basic definitions of mathematical morphology, which is also useful for the following chapters, and then proposed new methods of musical analysis and generation using mathematical morphology operators. The four main operations of morphology have been considered, and the relevance of each operation has been presented. In particular, we differentiate analysis operators, with erosion and opening, from generative operators, with dilation and closing. We have demonstrated that these morphological operators can be applied to different symbolic representations of music with pitch sets ($E = \mathbb{Z}$), musical rhythms ($E = Z_n$) and musical pieces ($E = \mathbb{R}^n$). One of the advantages of using mathematical morphology is to apply a musical property to musical data. In particular, the structuring element represents the property to be applied, such as major or minor if applied to a chord. This first approach reveals an original application of mathematical morphology, associating mathematical formulations (essentially algebraic) with musical data.

In this chapter, we have mainly demonstrated the relevance of morphological operators for musical analysis, i.e. erosion and opening. In particular, the erosion extracts specific musical characteristics, such as the root of a chord, the decomposition of a musical rhythm or the identification of chords in a piece of music, while the opening simplifies musical data, such as retaining only certain musical intervals of a chord, reducing a musical rhythm or removing some musical components from a piece. We also mentioned that generative morphological operators create or enrich

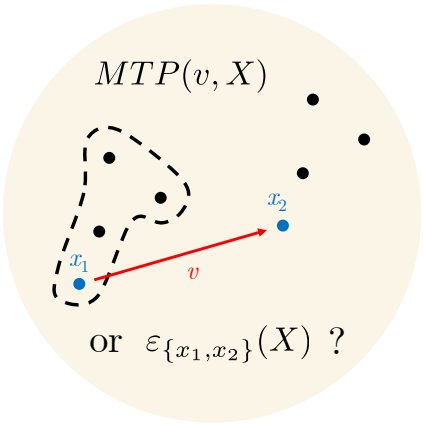
musical data. Dilation is a good example, as it generates a chord from the root and a musical property, for example. In addition, several musical properties can be applied in any order, as dilation is commutative. Closing enriches musical data, but is more complicated to apply, as it is a less intuitive, and we believe it is preferable to apply other operations to enrich data.

This chapter is limited to the basic operations of binary mathematical morphology, but there are plenty of other morphological operations (some of which have already been partially covered in this chapter). Therefore, a future work is to extend this research to find a musical meaning to other morphological operations. In addition, the theory of binary morphology can be generalized to grayscale morphology. Next, we demonstrate how to apply this generalization to similarity matrices in Chapter 4. In this situation, morphological filters help detect the main blocks of the similarity matrix, which is useful for discovering the structure of the musical piece.

While we have demonstrated in this chapter that morphological operators can be applied to musical data, we would like to point out that the results obtained are still very elementary. While this approach is original, these results can also be obtained without using mathematical morphology. However, we demonstrate in Chapter 2 and Chapter 3 that morphology can be used to obtain new results for the discovery of musical patterns. In particular, the use of known morphology results allows us to better understand and develop musical pattern discovery research. If this approach is successful, it is because the theory of mathematical morphology fits perfectly into the framework of musical pattern discovery, as we demonstrate in the next chapter.

Chapter 2

Links Between Point-Set Algorithms in Music and Mathematical Morphology



Contents

In this chapter, we review the main point-set algorithms for discovering and matching patterns in musical data. In particular, we demonstrate that mathematical morphology fits perfectly into this context, since the algorithms and definitions introduced can be easily described using morphological operators. Finally, we present how morphology can be used to better understand and develop point-set algorithms in music.

Section 2.1 presents the advantages of point-set algorithms in music. Section 2.2 reviews 9 point-set algorithms and provides links to mathematical morphology. Section 2.3 demonstrates that morphological erosion can generalize the maximal translatable pattern concept that is the foundation of point-set algorithms in music. Finally, Section 2.4 summarizes the main results presented in this chapter.

2.1 Introduction to Point-Set Algorithms in Music

Pattern discovery research in symbolic representation of music can be divided into two categories: methods based on a string representation and methods based on a multidimensional representation of music [Janssen 2013]. In this section we present the advantages and inconvenients of these two approaches, which sets the context for the next chapters.

2.1.1 String Representation of Music

The first attempts to discover repeating patterns in music were based on a string representation of music. In this case, music is represented as a string of symbols. From this representation, the goal is to find subsequences that repeat identically or nearly identically. This is usually achieved by defining a similarity between the elements of the string in order for the algorithm to extract the subsequences [Hsu 1998, Smith 2001]. Some methods for this were inspired by techniques developed in computational biology to compare gene sequences [Gusfield 1997]. The string representation is adapted for monophonic music or polyphonic music where each voice can be represented by a string, such as Bach's Fugues. In the case of Bach's Fugues, several algorithms have already been developed to discover the *subject* or the *counter-subject* (sometimes present, usually following the subject) [Giraud 2012b], the *episodes* (modulatory sections) [Giraud 2012a], the *stream segments* (small numbers of tones grouped together) [Rafailidis 2008], or the *global structure* [Giraud 2015]. However, string-based approaches present several weaknesses. First of all, converting music to a string representation is not always rele-

vant. For example, with a string-based approach, it can be very difficult to discover patterns in unvoiced polyphonic music (such as piano music). Then, when a pattern is embellished, with an extra note in the middle of the pattern for instance, the subsequence is divided in two and therefore remains undiscovered. Also, computation time can be very long unless some parameters are imposed, such as the maximum and minimum length of a pattern, which cannot be generalized directly to other musical data. For these three reasons, we have decided to focus on a multidimensional representation of music in the following.

2.1.2 Multidimensional Representation of Music

To avoid the problems associated with string-based approaches to discover musical patterns, Meredith et al. have developed a method for discovering patterns using a multidimensional representation of music [Meredith 2002a]. In this case, musical data are represented in $\mathbb{R}^n$, where each point corresponds to a musical note and each component of the space corresponds to a characteristic of the note (such as its note onset, its pitch, its duration, etc.). Based on this representation, a number of algorithms have been developed, called *point-set algorithms*, which are reviewed in Section 2.2. These algorithms are all inspired by the SIA algorithm, which computes the vectors between notes to obtain the largest translatable patterns in the musical data. In this way, these methods solve the problems associated with string-based approaches. However, the main weakness of these methods is to interpret the discovered patterns, as it produces too many repeating patterns, of which the majority are not musically interesting.

In the following we present some ideas that have been proposed to develop and improve the discovery of patterns using a multidimensional representation of music. In addition, we propose the use of mathematical morphology to describe and better understand some of these concepts. We will not recall the useful definitions and propositions of mathematical morphology in this chapter, as they have already been presented in the previous chapter in Section 1.2.

2.2 Links Between Point-Set Algorithms for Pattern Discovery and Pattern Matching in Music and Mathematical Morphology

In this section, we review the main point-set algorithms for the discovery and matching of musical patterns and make connections with mathematical morphology. In particular, we demonstrate that these algorithms can be simply described using the principal operators of mathematical morphology. Geoffray Boras Delattre helped develop some of the results presented here during a 3-week internship that I supervised.

2.2.1 The SIA Algorithm

The "Structure Induction Algorithm" (SIA) is the basis of point-set algorithms for discovering musical patterns. This algorithm was first proposed by Meredith et al. [Meredith 2001, Meredith 2002a]. The task of this algorithm is to discover the largest translated patterns, called the *Maximal Translatable Patterns* (MTPs), in a multidimensional dataset X. Here, the datasets are referred to by the letter X, as opposed to D used by Meredith et al., we choose this to better match the mathematical definitions of morphology.

Definition 2.1: Maximal Translatable Pattern (MTP)

Given a vector $v \in E$, the **MTP** for v in X is defined as follows:

$$MTP(v, X) = \{x \in X \mid x + v \in X\}$$

Instead of computing $MTP(v, X)$ for all $v \in E$, it is possible to optimize the MTP computations. Meredith et al. noticed that $MTP(v, X)$ is non-empty if and only if there are two points $x_1, x_2 \in X$ such that $v = x_2 - x_1$. Therefore, it is sufficient to compute the MTPs for the vectors $v = x_2 - x_1$ with $x_1, x_2 \in X$. In this case, v can be seen as the vector going from x_1 to x_2. So $-v$ is the vector from x_2 to x_1 and we do not need to compute $MTP(-v, X)$ as it is equal to $MTP(v, X)$ translated by v. Consequently, to avoid redundancy and compute $MTP(-v, X)$ as well, we only need to compute MTPs of the form:

$$MTP(x_2 - x_1, X) \text{ such that } x_1, x_2 \in X \wedge x_1 \leq x_2,$$

where $x_1 \leq x_2$ means that the lexicographic order of x_1 is lower than that of x_2. An example of $MTP(v, X)$ is displayed in Figure 2.1, where $v = x_2 - x_1$ is also shown in the figure with $x_1, x_2 \in X$.

For $x_1, x_2 \in X$, it is possible to express $MTP(x_2 - x_1, X)$ in a very simple form with the morphological erosion using the following lemma.

Lemma 2.1: $MTP(x_2 - x_1, X)$

Let $x_1, x_2 \in X$ and $\{x_1, x_2\}$ the structuring element with its origin set on x_1, we have:

$$MTP(x_2 - x_1, X) = \varepsilon_{\{x_1, x_2\}}(X)$$

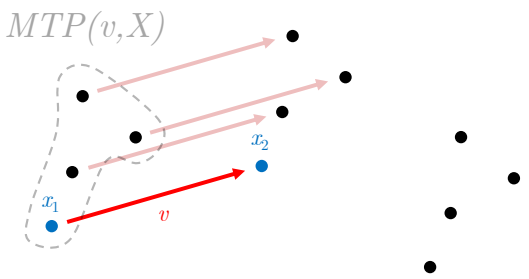


Figure 2.1: Illustration of the result of a MTP for a certain vector $v = x_2 - x_1$ with $x_1, x_2 \in X$. The set X is represented by the dots.

Proof of Lemma 2.1. Under the assumptions of the lemma, we have:

$$\begin{aligned} MTP(x_2 - x_1, X) &= \{x \in X \mid x + (x_2 - x_1) \in X\} \\ &= \{x \in X \mid x \in X \wedge x + (x_2 - x_1) \in X\} \\ &= \{x \in E \mid x \in X \wedge x + (x_2 - x_1) \in X\} \\ &= \{x \in E \mid (x_1 - x_1) + x \in X \wedge (x_2 - x_1) + x \in X\} \\ &= \{x \in E \mid \{x_1, x_2\}_x \subseteq X\} \\ &= \varepsilon_{\{x_1, x_2\}}(X) \end{aligned}$$

□

Finally, for a given dataset X, the SIA algorithm computes all the translations (i.e. $x_2 - x_1$ with $x_1, x_2 \in X$) and all the translated points (i.e. $MTP(x_2 - x_1, X)$). In other words, SIA computes the set:

$$\text{SIA} : \{< x_2 - x_1, MTP(x_2 - x_1, X) > \mid x_1, x_2 \in X \wedge x_1 \leq x_2\}$$

which can be expressed using morphological operations:

$$\text{SIA} : \{< x_2 - x_1, \varepsilon_{\{x_1, x_2\}}(X) > \mid x_1, x_2 \in X \wedge x_1 \leq x_2\}$$

2.2.2 The SIATEC Algorithm

SIA discovers the largest translated patterns given a translation, but provides no information on the other transposed occurrences of the discovered pattern. For example, in Figure 2.1, the pattern discovered by $MTP(v, X)$ is repeated on the right of the figure, and SIA does not indicate this. It would be useful if, in addition to discovering patterns, the algorithm could also reveal the repetitions of this pattern in the dataset. To obtain this information, Meredith et al. proposed the SIATEC algorithm [Meredith 2001, Meredith 2002a, Meredith 2006].

First, they defined the *Translation Equivalence Class* (TEC) of a pattern P by the set of patterns from X that are equal to P up to a translation.

Definition 2.2: Translation Equivalence Class (TEC)

Let $P \in \mathcal{P}(E)$, the **TEC** for P in X is defined as follows:

$$TEC(P, X) = \{Q \in \mathcal{P}(E) \mid \exists t \in E \wedge P_t = Q \wedge Q \subseteq X\}$$

As before, this definition can be expressed using morphological operators.

Lemma 2.2: $TEC(P, X)$

Let $P, X \in \mathcal{P}(E)$, we have:

$$TEC(P, X) = \{\delta_t(P) \in \mathcal{P}(E) \mid t \in \varepsilon_P(X)\}$$

Proof of Lemma 2.2. Under the assumptions of the lemma, we have:

$$\begin{aligned} TEC(P, X) &= \{Q \in \mathcal{P}(E) \mid t \in E \wedge P_t = Q \wedge Q \subseteq X\} \\ &= \{P_t \in \mathcal{P}(E) \mid t \in E \wedge P_t \subseteq X\} \\ &= \{\delta_t(P) \in \mathcal{P}(E) \mid t \in E \wedge P_t \subseteq X\} \\ &= \{\delta_t(P) \in \mathcal{P}(E) \mid t \in \varepsilon_P(X)\} \end{aligned}$$

□

Moreover, the TEC of a pattern P in X can also be expressed as a pair $< P, T(P, X) >$ consisting of the pattern P and its translators $T(P, X)$ defined as follows.

Definition 2.3: Translators $T(P, X)$

Let $P, X \in \mathcal{P}(E)$, the **set of translators** of P in X is defined as follows:

$$T(P, X) = \{t \in E \mid P_t \subseteq X\}$$

Thus, the TEC of a pattern can be obtained by translating P by its translators $t \in T(P, X)$. For example, Figure 2.2 displays the TEC and the set of translators of the pattern discovered in Figure 2.1. Once again, there is a direct link with the morphological erosion.

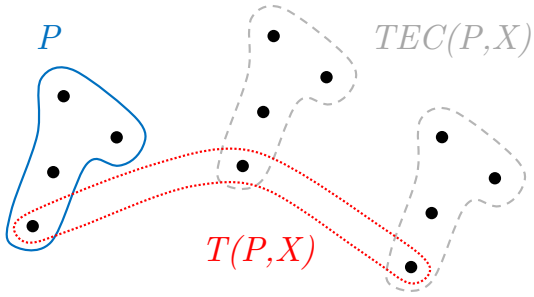


Figure 2.2: Illustration of the TEC and the set of translators of the pattern discovered in Figure 2.1. Note that the P pattern is included in the TEC. The set X is represented by the dots.

Lemma 2.3: $T(P, X)$

Let $P, X \in \mathcal{P}(E)$, we have by definition:

$$T(P, X) = \varepsilon_P(X)$$

Therefore, SIATEC uses all the patterns discovered with SIA and computes the TECs associated with these patterns. TECs can be summarized as a pattern and all its translators. Finally, for a given dataset X, SIATEC computes the largest translatable patterns P in X (i.e. the MTPs) and the set of translators of P in X (i.e. the $T(P, X)$). As with the SIA algorithm, this needs not be computed for all translations $v \in E$, but only those of the form $x_2 - x_1$ with $x_1, x_1 \in X$ and $x_1 \leq x_2$.

$$\text{SIATEC} : \{< MTP(x_2 - x_1, X), T(MTP(x_2 - x_1, X), X) > | \; x_1, x_2 \in X \wedge x_1 \leq x_2\}$$

Which can also be expressed using morphological operations:

$$\text{SIATEC} : \{< \varepsilon_{\{x_1, x_2\}}(X), \varepsilon_{\varepsilon_{\{x_1, x_2\}}(X)}(X) > | \; x_1, x_2 \in X \wedge x_1 \leq x_2\}$$

2.2.3 The SIACT Algorithm

The aim of the SIACT algorithm is to avoid having a pattern $P' \subseteq P$ that is musically more important than P. For example, by adding two points v apart to the X dataset defined in Figure 2.1, we obtain the set $MTP(v, X)$ with an additional point as shown in Figure 2.3. In this figure, $MTP(v, X)$ is made up of the four points on the left, as in Figure 2.1, and an additional point on the right. In particular, this point is isolated temporally from the rest of the pattern (note that the same reasoning can be applied to dimensions other than time). Thus, this new discovered MTP is not musically interesting, as it is perceived as the previous MTP to which

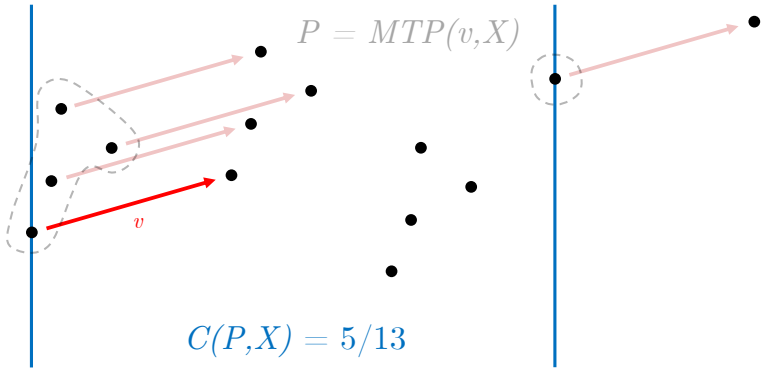


Figure 2.3: Illustration of low compactness value of a pattern discovered by a MTP. However, by selecting only the first four notes of the MTP, we obtain a pattern with a maximum compactness value equal to $4/4 = 1$.

an unrelated and completely detached note has been added. This example reveals that a MTP discovered by SIA can be broken down into several sub-patterns that are musically more meaningful. Collins et al. named this situation the "problem of isolated membership" [Collins 2010]. To solve this problem, they proposed the SIACT algorithm, which consists in computing SIA and then applying a "Compactness Trawler" for each MTP discovered, thus extracting sub-patterns from the MTPs that are more interesting from a musical point of view. The compactness trawler computes the compactness of MTP sub-patterns. The *Compactness* $C(P, X)$ of a pattern P in X is defined as the ratio of the number of points of P to the number of points of X in the region of P [Meredith 2002a]. For example, the compactness value for $MTP(v, X)$, shown in Figure 2.3, is 5/13 because $MTP(v, X)$ contains 5 points while there are 13 points of X in the region defined by $MTP(v, X)$. In this case, the compactness value is too low and $MTP(v, X)$ is not considered a pattern in its entirety. To determine whether a sub-pattern of a MTP is considered as a pattern, Collins et al. stated that it must contain at least b notes (which they set equal to 3), and have a compactness value equal to at least a (which they set equal to 2/3). They therefore defined two threshold values: compactness threshold a and point threshold b. Sub-patterns not satisfying these two threshold conditions are then deleted. Consequently, the SIACT algorithm provides musically interesting patterns from the MTPs discovered by SIA.

2.2.4 The SIAR Algorithm

With the aim of improving the precision and efficiency of SIA, Collins proposed the "Structure Induction Algorithm for r subdiagonals" (SIAR) [Collins 2011]. This algorithm only computes a subset of the MTPs produced by SIA. Collins considers SIAR to be equivalent to using a sliding window of size r in SIA [Collins 2010, Collins 2011]. Therefore, the additional condition $x_2 - x_1 \leq r$ in SIA is sufficient

to obtain SIAR, where $x_2 - x_1 \leq r$ means that r is the maximum allowed difference between the positions of x_1 and x_2 in a lexicographic order. Therefore, SIAR computes the set:

$$\text{SIAR} : \{< x_2 - x_1, \varepsilon_{\{x_1,x_2\}}(X) > |\ x_1, x_2 \in X \wedge 0 \leq x_2 - x_1 \leq r\}$$

Finally, SIAR can be used instead of SIA in the SIACT algorithm to obtain SIARCT.

2.2.5 The COSIATEC Algorithm

The aim of the COSIATEC algorithm, which stands for "COmpression with SIATEC", is to cover the entire dataset with a set of relevant non-overlapping TECs. As a reminder, a TEC $T = TEC(P, X)$ is composed of the patterns of X that are equal to P up to a translation. The original version of the COSIATEC algorithm was described by Meredith et al. [Meredith 2002a, Meredith 2006]. Before presenting this algorithm, we first define a few concepts to characterize the relevance of a TEC in a dataset, which is useful for COSIATEC. Once again, these concepts can be expressed with the basic operators of mathematical morphology. First, we define the *Covered Set* $COV(T, X)$ of T in X by the set of notes that T covers in X [Meredith 2013].

Definition 2.4: Covered set $COV(T, X)$

Let $P, X \in \mathcal{P}(E)$ and $T = TEC(P, X)$, the **covered set** $COV(T, X)$ of T in X is defined as follows:

$$COV(T, X) = \bigcup_{P \in T} P$$

This can be expressed using the morphological opening, one of the four other basic operators of mathematical morphology.

Lemma 2.4: $COV(T, X)$

Let $P, X \in \mathcal{P}(E)$ and $T = TEC(P, X)$, we have by definition:

$$COV(T, X) = \gamma_P(X)$$

Then, the *Coverage* of $T = TEC(P, X)$ in X is defined by the number of points in X that are part of the occurrences of T in X [Meredith 2002a]. That is, the coverage of T in X is equal to $|COV(T, X)|$, or $|\gamma_P(X)|$ in morphological terms. In addition, we also define the *Compression Ratio* $CR(T, X)$ of $T = TEC(P, X)$ in X,

which measures how relevant a pattern P is in X, in the sense that P occurrences do not overlap and are repetitive in X [Meredith 2013].

Definition 2.5: Compression Ratio $CR(T, X)$

Let $P, X \in \mathcal{P}(E)$ and $T = TEC(P, X)$, the **compression ratio** $CR(T, X)$ of T in X is defined as follows:

$$CR(T, X) = \frac{|COV(T,X)|}{|P|+|T(P,X)|-1}$$

Using Lemma 2.3 and Lemma 2.4, we can deduce that the compression ration can also be expressed with morphological operators with the following lemma.

Lemma 2.5: $CR(T, X)$

Let $P, X \in \mathcal{P}(E)$ and $T = TEC(P, X)$, we have by definition:

$$CR(T, X) = \frac{|\gamma_P(X)|}{|P|+|\varepsilon_P(X)|-1}$$

The compression ratio emphasizes pattern occurrences of a TEC that do not overlap in the dataset. Because the more P occurrences do not overlap in X, the greater $|COV(T, X)|$ is, $|P|$ and $|T(P, X)|$ do not change, so $CR(T, X)$ is greater if P occurrences do not overlap in X. In addition, the compression ratio also emphasizes patterns that repeat a lot in X, because the more occurrences of P in X, the greater $|COV(T, X)|$ and $|T(P, X)|$ are. Assuming that the occurrences of P in X overlap only slightly or not at all, $|COV(T, X)|$ increases faster than $|T(P, X)|$ when there are more occurrences of P in X. Thus, $CR(T, X)$ is greater when there are more occurrences of P in X. In this way, the compression ratio emphasizes patterns that do not overlap and that repeat a lot in X, which is the case for musically relevant patterns.

With these definitions, we can now describe the main steps in the COSIATEC algorithm. First, the SIATEC algorithm is applied to obtain the set of all TECs of X. The algorithm then selects the "best" TEC. The criterion for choosing the best TEC is defined in terms of the highest compression ratio (compactness or coverage can also be used). Next, all the points covered by the "best" TEC are removed from the dataset, i.e. X becomes equal to $X \setminus COV(T, X)$. From then on, as long as there are still points in X, we repeat the algorithm loop: apply SIATEC to X, select the "best" TEC and remove the covered set of the "best" TEC. The algorithm stops when X is empty. In this way, COSIATEC produces a family of TEC $\mathcal{F} = (T_i)_{i \in I}$

that cover the dataset X without overlap, i.e.:

$$X = \bigcup_{T_i \in \mathcal{F}} COV(T_i, X) \text{ such that } i \neq j \Rightarrow COV(T_i, X) \cap COV(T_j, X) = \emptyset$$

In morphological terms, with $T_i = TEC(P_i, X) \in \mathcal{F}$, this can be translated as:

$$X = \bigcup_{T_i \in \mathcal{F}} \gamma_{P_i}(X) \text{ such that } i \neq j \Rightarrow \gamma_{P_i}(X) \cap \gamma_{P_j}(X) = \emptyset$$

2.2.6 The SIATECCompress Algorithm

The aim of the SIATECCompress algorithm is to produce a set of TECs that covers the X dataset, but unlike COSIATEC, the covered sets of the TECs may overlap. Since COSIATEC runs SIATEC at each new iteration of the algorithm loop, this can be time-consuming on large datasets. To solve this problem, SIATECCompress runs SIATEC just once and selects all the best TECs sufficient to cover the dataset [Meredith 2013, Meredith 2015]. The selection criteria used to classify the best TECs are the same as for the COSIATEC algorithm (compression ratio, compactness or coverage). The difference is that the TECs are not computed again in the algorithm loop, so the ranking of the TECs according to the selection criterion always remains the same. Since the best TECs may have shared points, and the ranking of the best TECs is not updated, it is possible for a point to belong to several TECs covering the dataset. Therefore, unlike COSIATEC, overlaps are allowed, which means that SIATECCompress does not produce an encoding as compact as COSIATEC. Basically, SIATECCompress results are not as good as COSIATEC, but faster. Finally, like COSIATEC, the SIATECCompress algorithm ends when the entire dataset is covered by selected TECs. Thus, SIATECCompress produces a family of TECs $\mathcal{F} = (T_i)_{i \in I}$ that covers the dataset, formally: $X = \bigcup_{T_i \in \mathcal{F}} COV(T_i, X)$, or in morphological terms with $T_i = TEC(P_i, X)$: $X = \bigcup_{T_i \in \mathcal{F}} \gamma_{P_i}(X)$.

2.2.7 Forth's Algorithm

Forth's algorithm produces a set of TECs that covers the dataset, but the covered sets of these TECs may overlap and they may not completely cover the entire dataset, unlike the COSIATEC and SIATECCompress algorithms [Forth 2009, Forth 2012]. First, Forth's algorithm computes SIATEC to obtain all the TECs of the dataset. As with SIATECCompress, SIATEC is computed just once. Next, Forth assigns a weight to each TEC to determine its relevance. Unlike COSIATEC or SIATECCompress, it uses not just compression ratio or compactness separately, but mixes the two to define the weight $W(T)$ of a TEC T, which is useful for defining the "best" TEC:

$$W(T) = W_{CR}(T) \times W_{Comp}(T),$$

where $W_{CR}(T)$ is the normalized compression ration of T and $W_{Comp}(T)$ is the maximum value of the within-voice segment compactness of all the patterns in T, also

normalized. In the case where the dataset does not have separate voices, Meredith pointed out that within-voice segment compactness is almost equivalent to the bounding-box compactness [Meredith 2015]. Bounding-box compactness consists in calculating the compactness of a pattern on the smallest rectangular region that contains the pattern (for data in $\mathbb{R}^2$) and does not require the dataset to have separate voices.

Therefore, the algorithm iteratively selects the "best" TEC T by choosing the one that maximizes:

$$c \times W(T),$$

where c is the number of new points that T adds to the covered set of TECs already selected. In addition, to avoid having covered sets with too many overlaps, it is required that $c > c_{min}$ with c_{min} a threshold value which ensures that the new TEC adds at least c_{min} new points to the covered set of TECs already selected. Finally, if there are no new TECs to add, even if the covered set of TECs already selected does not completely cover the dataset, the algorithm ends. As a result, the Forth algorithm produces a family of TEC $\mathcal{F} = (T_i)_{i \in I}$ that partially covers the dataset, formally: $X \supseteq \bigcup_{T_i \in \mathcal{F}} COV(T_i, X)$, or in morphological terms with $T_i = TEC(P_i, X)$: $X \supseteq \bigcup_{T_i \in \mathcal{F}} \gamma_{P_i}(X)$.

2.2.8 The SIAM Algorithm

In addition to producing the foundations of point-set algorithms for discovering musical patterns, Meredith et al. also proposed the "SIA Matching" algorithm (SIAM), a pattern-matching algorithm based on SIA [Meredith 2001, Wiggins 2002, Meredith 2002a][1]. This algorithm can also be described with mathematical morphology operators, as we are about to demonstrate. The aim of this algorithm is to find the maximal matches of a pattern P in a multidimensional dataset X [Meredith 2002b], which is defined as follows.

Definition 2.6: Maximal match $MM(v, P, X)$

Let $P, X \in \mathcal{P}(E)$ and $v \in E$, the **maximal match** $MM(v, P, X)$ of P in X under translation v is defined as follows:

$$MM(v, P, X) = \{x \in P \mid x + v \in X\}$$

It is not necessary to compute $MM(v, P, X)$ for all $v \in E$, in particular it is sufficient to consider translations from a point of P to a point of X, i.e. $v = x_2 - x_1$

[1] Note that there is some confusion in the definitions of the SIAM, SIAME and SIAMESE algorithms and we have chosen here, in consultation with David Meredith, not to make any differences between these algorithms and to retain the main idea under the name of the SIAM algorithm.

with $x_1 \in P$ and $x_2 \in X$. In this way, SIAM returns the set of all maximal matches of P in X:

$$\text{SIAM} : \{< x_2 - x_1, MM(x_2 - x_1, P, X) >| \; x_1 \in P \wedge x_2 \in X\}$$

Ukkoven has already noted that maximal matches consist of a translation followed by an intersection [Ukkonen 2003], which can be seen as a morphological dilation by a vector followed by an intersection.

Lemma 2.6: $MM(x_2 - x_1, P, X)$

Let $P, X \in \mathcal{P}(E)$, $x_1 \in P$, $x_2 \in X$ and $\{x_2, x_1\}$ the structuring element with its origin set on x_2, we have:

$$MM(x_2 - x_1, P, X) = P \cap \delta_{\{x_2, x_1\}}(X)$$

Meredith et al. pointed out that an MTP can be viewed as a special case of a maximal match where $P = X$, i.e. $MTP(v, X) = MM(v, X, X)$, which leads to the morphological equality $\varepsilon_{\{x_1, x_2\}}(X) = X \cap \delta_{\{x_2, x_1\}}(X)$. Finally, SIAM can be rewritten with the morphological dilation as follows:

$$\text{SIAM} : \{< x_2 - x_1, P \cap \delta_{\{x_2, x_1\}}(X) >| \; x_1 \in P \wedge x_2 \in X\}$$

The result of the SIAM algorithm is illustrated in Figure 2.4, for P displayed in Figure 2.4(a), and X in Figure 2.4(b). The different lines, of different colors and textures, indicate complete or partial matching between P and X. For example, the solid red line is the complete matching (i.e. when $P' = P$).

(a) Pattern P (b) Dataset X

Figure 2.4: Illustration of the SIAM algorithm for finding the exact and partial matching of P in X, both represented by the dots. Each colored dotted line indicates a sub-pattern of P of at least two points that has been matched in X without redundancy (adapted from Meredith et al. [Meredith 2002a]).

2.3 Maximal Translational Equivalence Classes

In this section, we analyze some definitions and theorems developed by Collins and Meredith to better understand the patterns discovered by point-set algorithms [Collins 2013]. In particular, we prove that these results can be understood and developed using basic morphological operators.

2.3.1 Definition of Maximal Translational Equivalence Classes

Collins and Meredith introduced another type of maximal repeating pattern called *Maximal Translational Equivalence Class* (MTEC) [Collins 2013].

Definition 2.7: Maximal TEC (MTEC)

Let $P, X \in \mathcal{P}(E)$ such that $|TEC(P, X)| = m$. P is defined as **MTEC** if:

$$\forall P' \in \mathcal{P}(E), P \subsetneq P' \Rightarrow |TEC(P', X)| < m$$

In other words, if the pattern is enriched with new points, then the number of new pattern occurrences necessarily decreases. This is why the pattern is called "maximal". For example, in Figure 2.5, the pattern P is not MTEC because $|TEC(P, X)| = 3$ and it is possible to add the point on the left and still have three occurrences of the pattern in the dataset. By adding the point on the left, the pattern becomes MTEC.

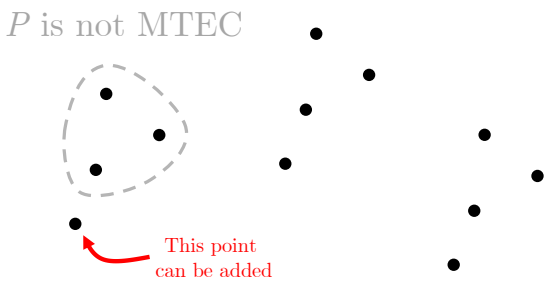


Figure 2.5: Illustration of a pattern P that is not MTEC, it is possible to add the point on the left without changing the number of three occurrences in X. The set X is represented by the dots.

2.3.2 MTEC Characterization Using Mathematical Morphology

The definition of an MTEC can be directly expressed with morphological erosion with the following:

$$\forall P' \in \mathcal{P}(E), P \subsetneq P' \Rightarrow |\varepsilon_{P'}(X)| < m$$

However, we can obtain a much better characterization of the notion of MTEC with morphological operators. Collins and Meredith have successfully characterized MTECs by an intersection of MTPs [Collins 2013]. Formally, for $P, X \in \mathcal{P}(E)$, they proved the following equivalence:

$$P \subseteq X \text{ is MTEC} \iff P = \bigcap_{v \in T(P,X)} MTP(v, X)$$

However, if $P \subsetneq X$ this equivalence is no longer true. For example, the pattern P can be included in X up to a translation. Musically, this means defining the pattern up to one harmonic or temporal transposition. Mathematically, this is equivalent to changing the origin of the pattern. Inspired by the previous equivalence, we propose to characterize an MTEC pattern in a simple way with Theorem 2.7.

Theorem 2.7: MTEC

Let $P, X \in \mathcal{P}(E)$, we have:

$$\boxed{P \text{ is MTEC} \iff P = \varepsilon_{\varepsilon_P(X)}(X)}$$

Proof of Theorem 2.7. Let $P, X \in \mathcal{P}(E)$ and $m \in \mathbb{N}$ such that $|\varepsilon_P(X)| = m$. This proof uses Lemma 3.1 and Theorem 3.5 (their proofs are given in the next chapter) which provide the following results:

$$\text{Lemma 3.1} : P \subseteq \varepsilon_{\varepsilon_P(X)}(X), \qquad \text{Theorem 3.5} : \varepsilon_P(X) = \varepsilon_{\varepsilon_{\varepsilon_P(X)}(X)}(X)$$

$\boxed{\Leftarrow}$ We prove this implication by contraposition, i.e. we have to prove that: P is not MTEC $\Rightarrow P \neq \varepsilon_{\varepsilon_P(X)}(X)$. Therefore, suppose that P is not MTEC. So there exists $P' \in \mathcal{P}(E)$ such that $P \subsetneq P'$ and $|\varepsilon_{P'}(X)| \geq m$. Setting that "decreasingness s.e." signifies decreasingness with respect to the structuring element, we have:

$$\begin{aligned}
& P \subseteq P' \\
\Rightarrow \quad & \varepsilon_{P'}(X) \subseteq \varepsilon_P(X) && \text{(decreasingness s.e. of } \varepsilon) \\
\Rightarrow \quad & \varepsilon_{P'}(X) = \varepsilon_P(X) && (|\varepsilon_P(X)| = m \text{ and } |\varepsilon_{P'}(X)| \geq m)
\end{aligned}$$

This allows us to conclude with the following implications:

$$
\begin{aligned}
& P' \setminus P \subseteq P' \\
\Rightarrow \quad & \varepsilon_{P'}(X) \subseteq \varepsilon_{P' \setminus P}(X) && \text{(decreasingness s.e of } \varepsilon) \\
\Rightarrow \quad & \varepsilon_{P}(X) \subseteq \varepsilon_{P' \setminus P}(X) && (\varepsilon_{P'}(X) = \varepsilon_P(X)) \\
\Rightarrow \quad & \varepsilon_{\varepsilon_{P' \setminus P}(X)}(X) \subseteq \varepsilon_{\varepsilon_P(X)}(X) && \text{(decreasingness s.e of } \varepsilon) \\
\Rightarrow \quad & P' \setminus P \subseteq \varepsilon_{\varepsilon_P(X)}(X) && (P' \setminus P \subseteq \varepsilon_{\varepsilon_{P' \setminus P}(X)}(X) \text{ Lemma 3.1}) \\
\Rightarrow \quad & P \neq \varepsilon_{\varepsilon_P(X)}(X) && (\exists p' \in P' : p' \notin P \wedge p' \in \varepsilon_{\varepsilon_P(X)}(X))
\end{aligned}
$$

$\boxed{\Rightarrow}$ We reason again by contraposition, i.e. we have to prove: $P \neq \varepsilon_{\varepsilon_P(X)}(X) \Rightarrow P$ is not MTEC. Thus, suppose that: $P \neq \varepsilon_{\varepsilon_P(X)}(X)$. Because Lemma 3.1 implies $P \subseteq \varepsilon_{\varepsilon_P(X)}(X)$, we have $P \subsetneq \varepsilon_{\varepsilon_P(X)}(X)$. Setting $P' = \varepsilon_{\varepsilon_P(X)}(X)$, we then have $P \subsetneq P'$ and the following equalities:

$$
\begin{aligned}
|\varepsilon_{P'}(X)| &= |\varepsilon_{\varepsilon_{\varepsilon_P(X)}(X)}(X)| \\
&= |\varepsilon_P(X)| && (\varepsilon_{\varepsilon_{\varepsilon_P(X)}(X)}(X) = \varepsilon_P(X) \text{ Theorem 3.5}) \\
&= m
\end{aligned}
$$

Therefore, we conclude that P is not MTEC because there exists $P' = \varepsilon_{\varepsilon_P(X)}(X)$ such that $P \subsetneq P'$ and $|\varepsilon_{P'}(X)| = m$. □

2.3.3 Generalization of MTPs using Morphological Erosion

Figure 2.6 illustrates the equivalence proved by Collins and Meredith where the MTEC pattern P is equal to an intersection of MTPs. In this figure, some points have been added to the set X defined in Figure 2.2, where the P pattern remains the same. We have already seen in Figure 2.5 that P is MTEC (if composed of four points). In this case, there are two non-zero translations v and v' from P into X, which are displayed in Figure 2.6. It can be observed that $MTP(v, X)$ and $MTP(v', X)$ are not equal to P because they have some extra points, as in the "problem of isolated membership" discussed in Section 2.2.3. However, with the intersection, we obtain:

$$P = MTP(v, X) \cap MTP(v', X)$$

Thus, intersections of MTPs seem to be a good solution to the "problem of isolated membership". This would avoid the computation of the compactness and provide MTECs patterns. However, without knowing $T(P, X)$, it is not possible to know which MTPs to intersect.

It is important to mention that Collins and Meredith made an interesting observation which connects the concepts of MTP and MTEC [Collins 2013]. They proved that:

$$\forall x_1, x_2 \in X, MTP(x_2 - x_1, X) \text{ is MTEC}$$

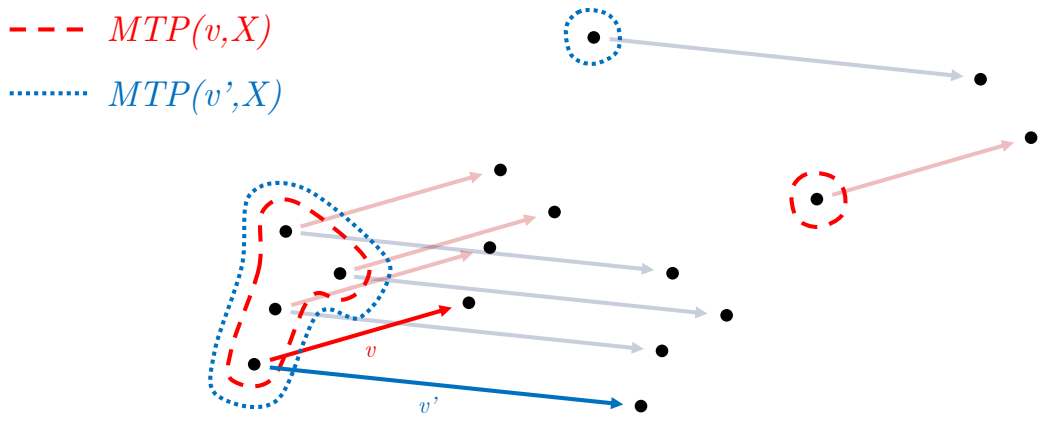


Figure 2.6: Illustration of an MTEC pattern P obtained with an intersection of MTPs, as $P = MTP(v, X) \cap MTP(v', X)$. The intersection removes isolated points and could be a solution to the "problem of isolated membership".

However they pointed out that the other way is not true. That is, a pattern P can be MTEC but without having two points $x_1, x_2 \in X$ which allow obtaining P with the MTP operation, formally:

$$\exists P \text{ MTEC } : \nexists x_1, x_2 \in X \text{ such that } P = MTP(x_2 - x_1, X)$$

Thanks to Theorem 2.7, we can understand this better. If the pattern P can not be obtained by two points $x_1, x_2 \in X$ with the MTP operation, this means that several points $x_1, \ldots, x_m \in X$ are required. But the MTP operation is not defined for several points $x_1, \ldots, x_m \in X$. This is why the morphological erosion of X by $x_1, \ldots, x_m \in X$ is required, rather than the MTP operation. We already proved that with two points $x_1, x_2 \in X$, we have: $MTP(x_2 - x_1, X) = \varepsilon_{\{x_1, x_2\}}(X)$ (Lemma 2.1). We then demonstrate that the erosion can be used to generalize the MTP concept, which is fundamental to point set algorithms as they are based on the discovery of MTPs with the SIA algorithm. Therefore, we have:

$$\forall P \text{ MTEC }, \exists x_1, \ldots, x_m \in X \text{ such that } P = \varepsilon_{\{x_1, \ldots, x_m\}}(X)$$

Moreover, Theorem 2.7 reveals how to obtain these points: $\{x_1, \ldots, x_m\} = \varepsilon_P(X)$. In this way, Corollary 2.8 reveals that a pattern is MTEC if and only if it is discovered by a morphological erosion.

Corollary 2.8: MTECs are patterns discovered by erosion

Let $P, X \in \mathcal{P}(E)$, we have:

$$P \text{ is MTEC} \iff \exists S \in \mathcal{P}(E) : P = \varepsilon_S(X)$$

Proof of Corollary 2.8. Let $P, X \in \mathcal{P}(E)$. We prove this corollary by double implication.
$\boxed{\Rightarrow}$ Suppose that P is MTEC, using Theorem 2.7 this means that $P = \varepsilon_{\varepsilon_P(X)}(X)$. By setting $S = \varepsilon_P(X)$, we have the existence of $S \in \mathcal{P}(E)$ such that: $P = \varepsilon_S(X)$.
$\boxed{\Leftarrow}$ Let $S \in \mathcal{P}(E)$ such that $P = \varepsilon_S(X)$. Using Theorem 2.7, it is sufficient to prove that $P = \varepsilon_{\varepsilon_P(X)}(X)$.

$$\begin{aligned} \varepsilon_{\varepsilon_P(X)}(X) &= \varepsilon_{\varepsilon_{\varepsilon_S(X)}(X)}(X) \\ &= \varepsilon_S(X) && (\varepsilon_{\varepsilon_{\varepsilon_S(X)}(X)}(X) = \varepsilon_S(X) \text{ Theorem 3.5}) \\ &= P \end{aligned}$$

□

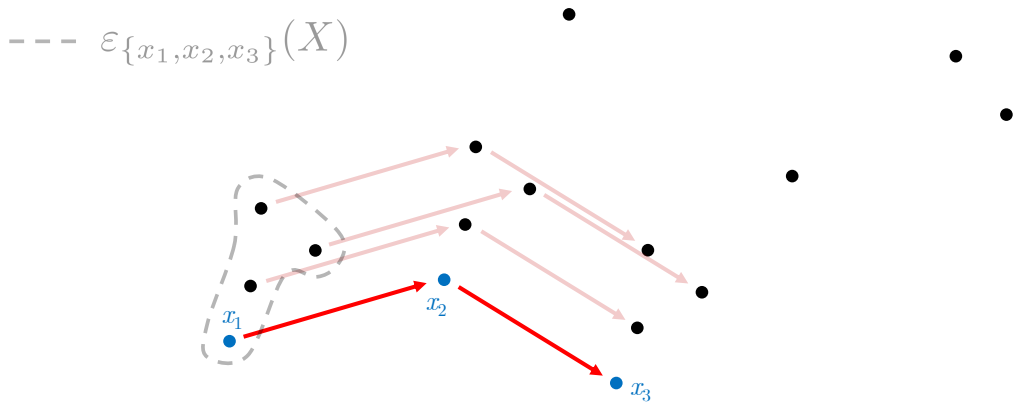


Figure 2.7: Illustration of an MTEC pattern P obtained with an erosion, as $P = \varepsilon_{\{x_1,x_2,x_3\}}(X)$. Therefore, erosion is equivalent to intersecting MTPs, as illustrated in Figure 2.6, and directly yields MTEC patterns.

This result is illustrated in Figure 2.7. Instead of interpreting the pattern P as the intersection of $MTP(v, X)$ and $MTP(v', X)$, as in Figure 2.6, Corollary 2.8 demonstrates that P can be interpreted as the erosion of X by $\{x_1, x_2, x_3\}$, where $\{x_1, x_2, x_3\} = \varepsilon_P(X)$. Therefore, we have:

$$P = \varepsilon_{\{x_1,x_2,x_3\}}(X)$$

Since the points $\{x_1, x_2, x_3\}$ are defined by $\varepsilon_P(X)$, it is a priori necessary to identify P in order to obtain them. This is the subject of the next chapter, where we propose a method of obtaining these points without having to know P. This theory reveals an original method of discovering musical patterns using morphological operators, and proposes mathematical results to support its effectiveness.

2.4 Summary

In Section 2.2, we reviewed the main point-set algorithms for pattern discovery and pattern matching in music. For an overview and links between these algorithms, Figure 2.8 summarizes the 9 algorithms described above. We have differentiated three different objectives of these algorithms: discover all repeating patterns (SIA/SIATEC/SIACT/SIAR/SIARCT), discover the most important patterns to describe the dataset (SIATECCompress/COSIATEC/Forth), find maximal matches for a pattern (SIAM).

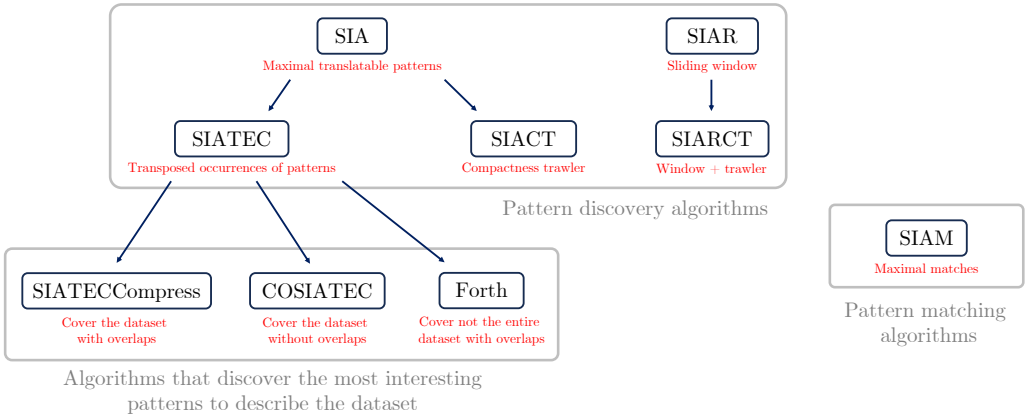


Figure 2.8: Summary of the links between the main point-set algorithms.

Some of these algorithms can be easily expressed using morphological operators. In particular, we summarize this in Table 2.1 with the first four algorithms originally developed by Meredith et al: SIA, SIATEC, COSIATEC and SIAM [Meredith 2002a].

Table 2.1: Principal algorithms for discovering repeated patterns in multidimensional dataset described with mathematical morphology operators.

Algorithms	**Description using mathematical morphology**
SIA	$\{< x_2 - x_1, \varepsilon_{\{x_1,x_2\}}(X) > \mid x_1, x_2 \in X \wedge x_1 \leq x_2\}$
SIATEC	$\{< \varepsilon_{\{x_1,x_2\}}(X), \varepsilon_{\varepsilon_{\{x_1,x_2\}}(X)}(X) > \mid x_1, x_2 \in X \wedge x_1 \leq x_2\}$
COSIATEC	$X = \bigcup_i \gamma_{P_i}(X)$ such that $i \neq j \Rightarrow \gamma_{P_i}(X) \cap \gamma_{P_j}(X) = \emptyset$
SIAM	$\{< x_2 - x_1, P \cap \delta_{\{x_2,x_1\}}(X) > \mid x_1 \in P \wedge x_2 \in X\}$

In addition to expressing some point-set algorithms with morphological operators, Table 2.2 shows that the majority of definitions used to discover patterns with point-set algorithms can also be expressed in terms of morphological operators.

Table 2.2: Principal definitions used for point-set algorithms for discovering patterns described with mathematical morphology operators.

Definition using mathematical morphology
$MTP(x_2 - x_1, X) = \varepsilon_{\{x_1,x_2\}}(X)$
$TEC(P, X) = \{\delta_t(P) \in \mathcal{P}(E) \mid t \in \varepsilon_P(X)\}$
$T(P, X) = \varepsilon_P(X)$
$COV(T, X) = \gamma_P(X)$
$CR(T, X) = \frac{\lvert\gamma_P(X)\rvert}{\lvert P\rvert+\lvert\varepsilon_P(X)\rvert-1}$
$MM(x_2 - x_1, P, X) = P \cap \delta_{\{x_2,x_1\}}(X)$
P is MTEC $\iff P = \varepsilon_{\varepsilon_P(X)}(X)$

However, the entire theory of point-set algorithms cannot be expressed exclusively in terms of morphological operators either. For example, the weight $W(T)$ of Forth's algorithm or the compactness of a pattern $C(P, X)$ cannot easily be expressed in morphological terms, or at least there is no point in doing so because it does not shorten the expression. Nevertheless, we believe that this characterization in morphological terms of Table 2.1 and Table 2.2 allows us to better understand some results and to extend the theory using the morphological result, as demonstrated in the next chapter.

2.5 Conclusion

In this chapter, we have reviewed the main point-set algorithms for pattern discovery and matching in music. In addition, we have demonstrated that mathematical morphology can describe these algorithms, and allows us to understand and develop some concepts related to repeating patterns in musical data. For example, we have revealed that morphological erosion generalizes the principle of MTP, which is the foundation of point-set algorithms. Furthermore, we have proven that MTEC patterns can be expressed very simply using mathematical morphology. Finally, since

we have proved that morphological erosion provides MTEC patterns, this might be a solution to a problem posed by Collins and Meredith when they stated that:

> "We need to gain a better understanding of how to discover MTEC patterns" [Collins 2013]

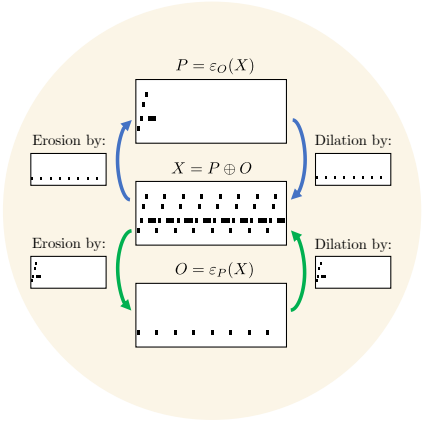
To do this, all we need to do is discover patterns using erosion, not MTPs. However, finding an appropriate structuring element to discover a musically interesting pattern is not an easy task. In the next chapter, we propose a method for discovering patterns from morphological erosion, focusing only on specific structuring elements. The originality of this approach lies in the musical interpretation of the discovered patterns and the chosen structuring elements. This avoids computing all the MTECs and focuses on those of musical interest only. Therefore, the next chapter responds to the request of Collins and Meredith for an algorithm that discovers the most relevant MTEC patterns in musical data:

> "Generating all the MTEC patterns for even a modestly sized piece of music would be impractical [...]. Designing a practical algorithm for generating only the perceptually salient MTEC patterns is therefore an interesting problem for future research. Once such an algorithm has been developed, it will be possible to explore more rigorously whether perceptually salient musical patterns correspond more closely to MTEC patterns or MTPs" [Collins 2013]

Finally, having established links between mathematical morphology and point-set algorithms provides a better understanding and development of research on musical pattern discovery, as well as mathematical results inspired by musical problems, as presented in the next chapter.

Chapter 3

Using Mathematical Morphology to Discover Repeated Patterns in Music



Contents

This chapter deals with the discovery of repeated patterns in symbolic representations of music using the theory of mathematical morphology. The main idea proposed here is to use the onsets to discover musical patterns. By definition, the morphological erosion of musical data by a musical pattern corresponds to its onsets. However, the erosion of musical data by the onsets is not always equal to the musical pattern. We propose a theorem which guarantees the equality if the musical pattern satisfies a topological condition. This condition is met when the patterns do not intersect, or only slightly, which is coherent in a musical context. Due to the importance of repetition in music, this idea proves to be relevant for the music pattern discovery task.

Section 3.1 presents the originality of the method proposed in this chapter and describes the context to discover musical patterns using the mathematical morphology theory. Section 3.2 provides the main result of this chapter with the proof that the morphological erosion of the onsets is equal to the musical pattern if it satisfies a specific topological condition which is coherent in a musical context. We then prove in Section 3.3 that if the pattern does not respect this condition, the erosion by its onsets completes the missing notes of the pattern required to obtain the equality between the pattern and the erosion by its onsets. Since there are usually several possible pairs of patterns and their onsets in musical data, we define a simple criterion in Section 3.4 to determine the pair that best represents the musical notions of a pattern and its onsets. Section 3.5 is dedicated to the application of the proposed method to symbolic representations of music, where the main musical patterns are discovered and result in a description of the musical piece by a union of morpholog-

ical dilation between patterns and their onsets. Finally, Section 3.6 concludes this chapter and proposes future work to apply this method in other musical situations.

3.1 Introduction

In this section, we describe the originality and the context of the method proposed in this chapter. In particular, we show that this approach is a continuation of existing musical pattern discovery algorithms.

3.1.1 Introduction to the Proposed Method

In this chapter, we propose to use the theory of mathematical morphology to discover repeated patterns in discrete sets of points. Even if we focus on symbolic representations of music, the developed results can be applied to other types of discrete set data composed of repeated patterns, defined as sets of discrete points starting at points interpreted as onsets in a musical context. While mathematical morphology has been widely applied to image processing, analysis and understanding, there are only few direct applications of this theory, in particular in its algebraic setting, to symbolic representations of music. This is partly due to the different nature of the patterns. On the one hand, the objects associated with digital image processing are defined on a discrete grid, endowed with a discrete connectivity, and are often defined as connected sets. On the other hand, those associated with symbolic music are sparse, in the sense that they are often not connected, according to the underlying connectivity of the space of representation. It is therefore necessary to adapt the morphological tools and the expected results when applying this theory to symbolic representations of music. This has been started by adapting the basic operators of mathematical morphology to obtain a musical meaning. In particular, we detailed in Chapter 1 how mathematical morphology can be applied to music for both analysis and generation. We also demonstrated in Chapter 2 that the theory of mathematical morphology fits perfectly into the framework of musical pattern discovery. In particular, morphological operators can describe and generalize existing concepts related to the discovery of repetitive patterns in musical data. Following these previous works, we advocate in this chapter that mathematical morphology can provide relevant tools for the discovery of repeated patterns in symbolic representations of music. In particular, we provide various mathematical results related to musical problems. These results optimize the discovery of musical patterns, and are the foundation for a new approach to discover musical patterns. The originality of this approach comes from the use of a musical meaning to discover patterns by distinguishing the role of onsets or musical pattern. Finally, the results developed in this chapter provide answers to problems arising from musical pattern discovery tasks, such as the development of an algorithm that discovers MTEC patterns, or on the musical interpretations and salience of patterns discovered by such an algorithm.

3.1.2 Context of the Proposed Method

In this chapter, mathematical morphology is applied to discover musical patterns. In particular, we consider here the simple case of binary mathematical morphology. We refer to Chapter 1 for definitions and properties of the principal operators of mathematical morphology, and to Chapter 2 for a review of the main musical pattern discovery algorithms. The method we propose here belongs to the category of algorithms that discover patterns in order to describe musical data, such as COSIATEC [Meredith 2002a], SIATECCompress [Meredith 2013] or Forth's algorithm [Forth 2009]. Therefore, in this chapter, the set E is equal to $\mathbb{R}^n$, the musical data X is a subset of E (i.e. $X \in \mathcal{P}(E)$), and the objective is to discover a family $\mathcal{F} = (P_i)_{i \in I}$ of musical patterns that describes X in the sense that:

$$\gamma_{\mathcal{F}}(X) \approx X$$

where the relation $\approx$ can be interpreted by a relatively high proportion of common points between $\gamma_{\mathcal{F}}(X)$ and X, patterns $P_i \in \mathcal{P}(E)$ are not singletons, I is an index set, and the opening by $\mathcal{F}$ is the union of the openings by the elements of $\mathcal{F}$ (as already described in Section 1.3.3), i.e.:

$$\gamma_{\mathcal{F}}(X) = \bigcup_{P_i \in \mathcal{F}} \gamma_{P_i}(X)$$

But how do we obtain the elements of $\mathcal{F}$? We propose here to exploit the property of morphological erosion as a generalization of MTPs, as described in Section 2.3.3. We demonstrate that, by selecting the appropriate structuring elements of the erosion, this operation discovers salient musical patterns. But how do we select the appropriate structuring elements to discover such musical patterns with the erosion? The results developed in this chapter prove that the structuring elements have to be the onsets of the patterns to be discovered. Since it can be easier to discover the onsets than the musical patterns from a computational point of view, this is an original approach for discovering musical patterns. In addition, as patterns are discovered with morphological erosion, this method discovers MTEC patterns and is therefore a solution to the problem identified by Collins and Meredith [Collins 2013], which is finding a method to discover salient MTEC patterns as described in Section 2.5. Consequently, based on the results of the theorems presented in this chapter, we propose an original method for discovering musical patterns using morphological erosion by the onsets.

3.2 Discovering the Musical Pattern From the Onsets

In this section, we propose a theorem that demonstrates that musical patterns can be discovered from their onsets if they satisfy a topological condition. This theorem reveals an original approach for discovering musical patterns in a multidimensional representation of music.

3.2.1 Presentation of the Problem of Discovering the Musical Pattern From its Onsets

We present here the problem of discovering the musical pattern from its onsets. We define the onsets of a pattern as the origins of its appearances in musical data. This makes the connection with the morphological erosion as described in Definition 3.1.

Definition 3.1: Onsets

Let $P, X \in \mathcal{P}(E)$. The **onsets** O of P in X are defined by:

$$O = \varepsilon_P(X)$$

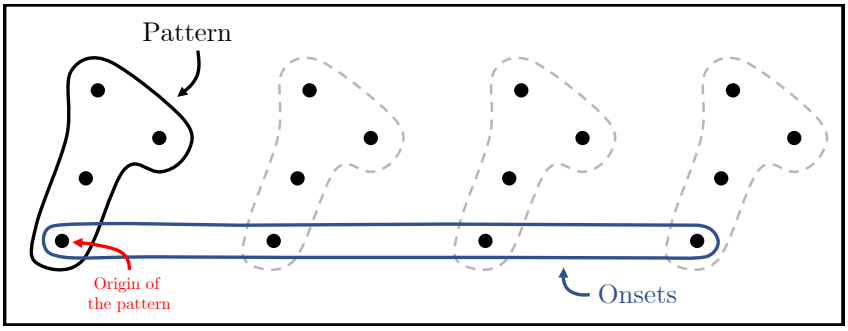
The onsets of a pattern can be interpreted as the beginnings of this pattern in the musical data if the origin of the pattern is placed on its first note, assuming that the dimensions of space are oriented. For example, in Figure 3.1(a), musical data are represented by dots, which can be seen as a pattern repeated four times. Repeats of the pattern are indicated in the figure by dotted lines. The onsets of this pattern are therefore composed of four dots, which correspond to the beginnings of the repeated pattern. The interesting result is illustrated in Figure 3.1(b) by now considering the onsets as a pattern: the onsets of the onsets of the pattern are equal to the pattern. In other words, by computing the points where the onsets occur, we discover the pattern (the repetitions of the onsets are indicated by the dotted lines in the figure). This result is quite surprising, and we can formalize it with morphological operators in Definition 3.2.

Definition 3.2: Discovering the musical pattern from its onsets

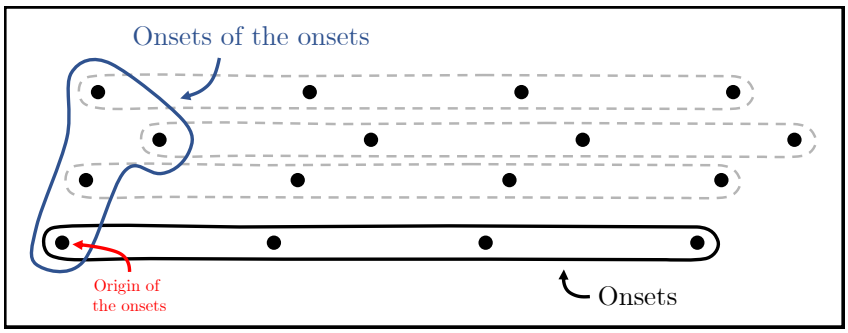
Let $P, O, X \in \mathcal{P}(E)$, where $O = \varepsilon_P(X)$. The problem of **discovering the musical pattern from its onsets** is to understand when the following equation is true:

$$P = \varepsilon_O(X)$$

Let $P, O, X \in \mathcal{P}(E)$ such that $O = \varepsilon_P(X)$. This can be interpreted as X the musical data, P a musical pattern and O its onsets. First of all, the problem of discovering the musical pattern from its onsets, i.e. the equality $P = \varepsilon_O(X)$, is not always true as shown in Figure 3.2. However, Lemma 3.1 allows us to state that the inclusion $P \subseteq \varepsilon_O(X)$ is always true. Therefore, P is always included in the onsets of O. This implies that taking the onsets of the onsets of a pattern enlarges it, as can be seen in Figure 3.2 where P, represented in Figure 3.2(b), is included



(a) Representation of the pattern and its onsets, which are interpreted as the points where the pattern begins. Pattern repeats are indicated by the dotted-line border. The origin of the pattern is chosen at the starting point of the pattern (assuming that the time axis is from left to right).



(b) The onsets of the onsets are equal to the pattern. Onsets repeats are indicated by the dotted-line border. That is, the dots corresponding to where the onsets begin are the same as the pattern dots. As before, the origin of the onsets is chosen at its starting point.

Figure 3.1: Presentation of the problem of discovering the pattern from its onsets. In this example, the onsets of the onsets of the pattern are equal to the pattern, which shows that it is sufficient to know the onsets to discover the pattern.

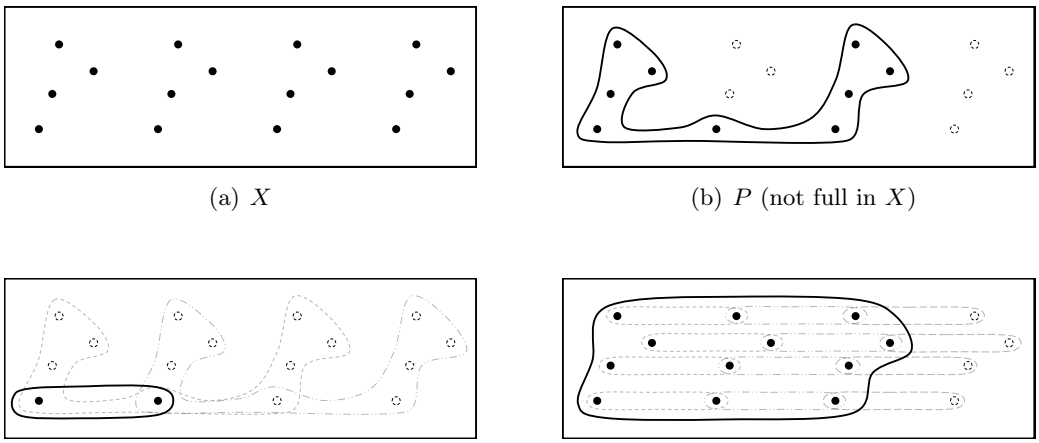
in $\varepsilon_O(X)$, represented in Figure 3.2(d). Note that the figures are 2D examples, for the sake of clarity, but all theoretical results apply for any finite dimension.

Lemma 3.1: $P \subseteq \varepsilon_O(X)$

Let $P, O, X \in \mathcal{P}(E)$, such that $O = \varepsilon_P(X)$ and $O \neq \emptyset$. We have:

$$P \subseteq \varepsilon_O(X)$$

Proof of Lemma 3.1. Let $p \in P$ and $o \in O$. Because $O = \varepsilon_P(X)$ and $O \neq \emptyset$, $\exists x \in X : p + o = x$. This is true for all $o \in O$, consequently $O_p \subseteq X$, which leads to $p \in \varepsilon_O(X)$. □



(a) X (b) P (not full in X)

(c) $O = \varepsilon_P(X)$ (d) $P \neq \varepsilon_O(X)$

Figure 3.2: Example where P is not full in X (Definition 3.3) and $P \neq \varepsilon_O(X)$, however the inclusion $P \subseteq \varepsilon_O(X)$ is satisfied (the origin is the first point on the left).

In the remainder of Section 3.2, we provide additional relations between P and O using morphological operators. In particular, we prove that the problem of discovering the musical pattern from its onsets can be solved under some assumptions that are musically interpretable.

3.2.2 Main Results

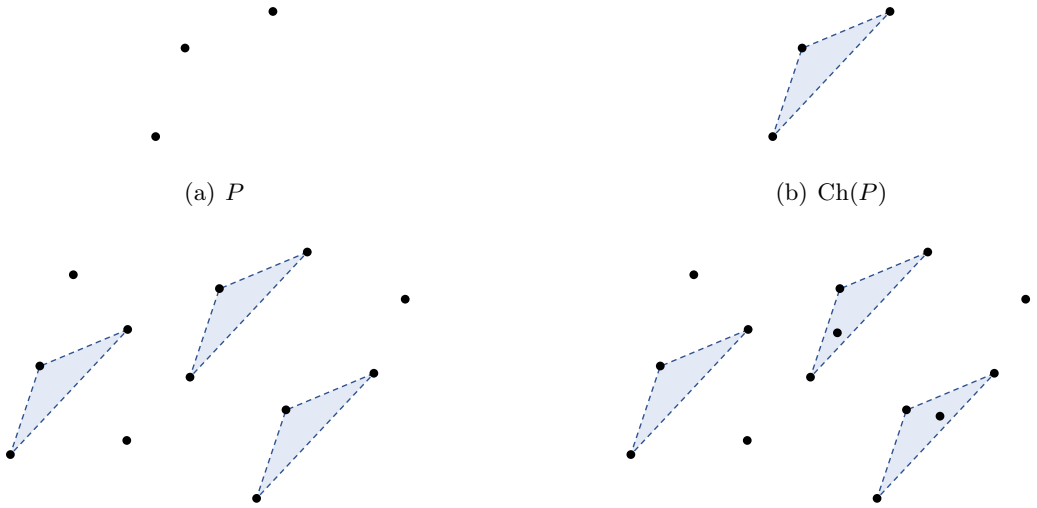
Because the equality $P = \varepsilon_O(X)$ is not always true, we need to add an additional condition on P and X to ensure this equality. We already know from Theorem 2.7 that this equality is satisfied if and only if P is MTEC. However, we propose another condition that could be interpreted musically and that would ensure this equality without using the MTEC property. To do this, we introduce a new definition, illustrated in Figure 3.3, that is, to some extent, the counterpart of the connectivity (without holes) used in image processing. For $P \in \mathcal{P}(E)$, let us note $\mathrm{Ch}(P)$ the *convex hull*[1] of P.

Definition 3.3: P full in X

Let $P, X \in \mathcal{P}(E)$. P is **full** in X if $P_t \subseteq X$ for any t implies that $\mathrm{Ch}(P_t)$ does not contain any point of X other than P_t, i.e.:

$$\forall t \in E, P_t \subseteq X \Rightarrow \mathrm{Ch}(P_t) \cap X = P_t$$

[1]Defined as: $\mathrm{Ch}(P) = \{\sum_{i=1}^{k} \lambda_i p_i \mid k \in \mathbb{N}^* \wedge \sum_{i=1}^{k} \lambda_i = 1 \wedge \forall 1 \leq i \leq k : p_i \in P \wedge \lambda_i \geq 0\}$.



(a) P (b) Ch(P)

(c) P is full in X (d) P is not full in X

Figure 3.3: Musical pattern P (a) and its convex hull (b). P is full in X (set of dots) in (c) because there are no dots inside the convex hull where it is included in X. However, this is not the case for (d).

Moreover, we consider the particular case where the data X is composed of patterns P which are repeated with translations, that is to say $\gamma_P(X) = X$. Under the condition that P is full in X and X is composed of solely of P patterns, Theorem 3.2 provides the equality $P = \varepsilon_O(X)$. Therefore, it is possible, under the assumptions of the theorem, to discover the pattern P from its onsets O.

Theorem 3.2: $P = \varepsilon_O(X)$

Let $P, X \in \mathcal{P}(E)$ such that:

- $\gamma_P(X) = X$,
- P is full in X,
- X is bounded.

Then, by defining $O = \varepsilon_P(X)$, we have:

$$P = \varepsilon_O(X)$$

We provide another link between P and O using morphological operators with the following lemma, which is useful for the proof of Theorem 3.2.

Lemma 3.3: $\varepsilon_O(X) = \varphi_O(P)$

Let $P, O, X \in \mathcal{P}(E)$, such that $O = \varepsilon_P(X)$ and $\gamma_P(X) = X$. We have:

$$\boxed{\varepsilon_O(X) = \varphi_O(P)}$$

Proof of Lemma 3.3. Under the assumptions of the lemma, we have:

$$\begin{array}{lll}
 & \gamma_P(X) = X & \\
\Rightarrow & \delta_P(\varepsilon_P(X)) = X & \text{(definition of } \gamma_P\text{)} \\
\Rightarrow & \delta_P(O) = X & \text{(definition of } O\text{)} \\
\Rightarrow & \delta_O(P) = X & \text{(commutativity of } \oplus\text{)} \\
\Rightarrow & \varepsilon_O(\delta_O(P)) = \varepsilon_O(X) & \text{(composing by } \varepsilon_O\text{)} \\
\Rightarrow & \varphi_O(P) = \varepsilon_O(X) & \text{(definition of } \varphi_O\text{)}
\end{array}$$

□

Proof of Theorem 3.2. Under the assumptions of the theorem, we can use Lemma 3.3 and Lemma 3.1. We prove that $P = \varepsilon_O(X)$ with two inclusions.

$\boxed{\subseteq}$ The first inclusion comes directly from Lemma 3.1.

$\boxed{\supseteq}$ It has been proved by Serra that for every $C \in \mathcal{P}(E)$ convex and $S \in \mathcal{P}(E)$ bounded, we have: $\varphi_S(C) = C$ (proposition IV-4 in [Serra 1982]). Since X is bounded, $O = \varepsilon_P(X)$ is also bounded and $\mathrm{Ch}(P)$ is convex by definition. Consequently, we have:

$$\varphi_O(\mathrm{Ch}(P)) = \mathrm{Ch}(P)$$

Let $t \in E$ such that $P_t \subseteq X$ (such t exists because $\gamma_P(X) = X$), therefore $t \in O$ and we have the following implications:

$$\begin{array}{lll}
 & P \subseteq \mathrm{Ch}(P) & \\
\Rightarrow & \varphi_O(P) \subseteq \varphi_O(\mathrm{Ch}(P)) & (\varphi_O \text{ is increasing}) \\
\Rightarrow & \varphi_O(P) \subseteq \mathrm{Ch}(P) & (\varphi_O(\mathrm{Ch}(P)) = \mathrm{Ch}(P)) \\
\Rightarrow & \varepsilon_O(X) \subseteq \mathrm{Ch}(P) & (\varphi_O(P) = \varepsilon_O(X) \text{ Lemma 3.3}) \\
\Rightarrow & \varepsilon_O(X)_t \subseteq \mathrm{Ch}(P)_t & \text{(translation by } t\text{)} \\
\Rightarrow & \varepsilon_O(X)_t \subseteq \mathrm{Ch}(P_t) & (\mathrm{Ch}(P)_t = \mathrm{Ch}(P_t)) \\
\Rightarrow & \varepsilon_{O_{-t}}(X) \subseteq \mathrm{Ch}(P_t) & (\varepsilon_O(X)_t = \varepsilon_{O_{-t}}(X)) \\
\Rightarrow & \varepsilon_{O_{-t}}(X) \cap X \subseteq \mathrm{Ch}(P_t) \cap X & \text{(intersection with } X\text{)} \\
\Rightarrow & \varepsilon_{O_{-t}}(X) \subseteq \mathrm{Ch}(P_t) \cap X & (\varepsilon_{O_{-t}}(X) \subseteq X \text{ because } O_E \in O_{-t}) \\
\Rightarrow & \varepsilon_{O_{-t}}(X) \subseteq P_t & (\mathrm{Ch}(P_t) \cap X = P_t \text{ because } P \text{ is full}) \\
\Rightarrow & \varepsilon_O(X) \subseteq P & \text{(translation by } -t\text{)}
\end{array}$$

□

3.2.3 Musical Interpretations of the Theorem

Theorem 3.2 is illustrated in Figure 3.4 with a piece of music X composed of a musical pattern P repeated several times (therefore $\gamma_P(X) = X$). This figure uses a piano roll representation, where the x-axis represents the time and the y-axis the pitch of the notes. The assumptions of Theorem 3.2 are satisfied because X is bounded and P is full in X. The morphological erosion of X by the musical pattern P is equal to the onsets O. Moreover, the theorem ensures that the other way around is true: we can obtain the pattern P by applying the erosion by the onsets O. All these morphological links between the piece X, the pattern P and its onsets O are summarized in Figure 3.4. In this case, we can describe the piece of music X by a morphological dilation between the pattern and its onsets:

$$X = P \oplus O$$

This last equality increases the relevance of the use of morphological operators applied to music.

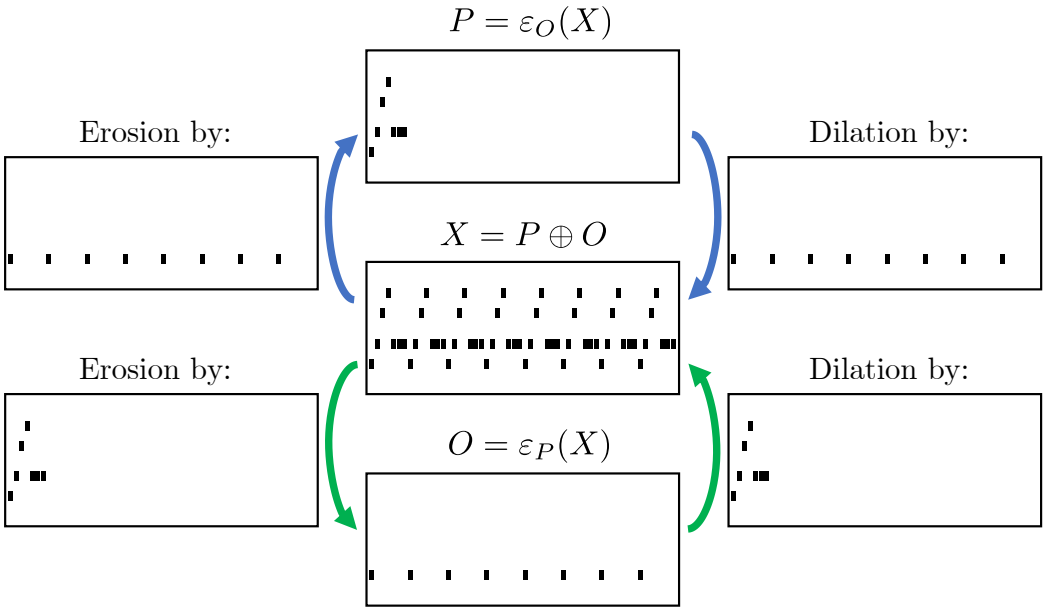


Figure 3.4: Illustration using morphological operators of the relations between the musical pattern P, which is full in X, and its onsets O.

Remark (About the assumptions in Theorem 3.2). In Theorem 3.2, there are three assumptions which are: P is full in X, $\gamma_P(X) = X$ and X is bounded.

- P is full in X: This assumption is coherent in a musical context. In the *Generative Theory of Tonal Music* (GTTM), the authors state several rules about overlaps in music [Lerdahl 1985]. In particular, the *Grouping Well-Formedness*

Rules 4 asserts:

> "If a group G_1 contains part of a group G_2, it must contain all of G_2"

This proves that it is very rare for patterns to intersect in music. However, later in GTTM, in Section *3.4 Grouping Overlaps*, it is mentioned that this rule is not always true and that patterns can overlap in a specific case: if patterns overlap, it is only the first or last note of the pattern that can be part of two patterns. Therefore, if we assume that a pattern P satisfies the GTTM rules (i.e. patterns do not intersect or only share their first or last note) and that the data X is composed of P which is repeated with translations (i.e. $\gamma_P(X) = X$), this leads to having a pattern P that is full in X. This reasoning shows that the property that a pattern is full is perfectly coherent in a musical context. Moreover, this property allows us to obtain a fundamental result with the Theorem 3.2, and if we remove this assumption, the theorem is not always true, as shown in Figure 3.2.

- $\gamma_P(X) = X$: In general, there are several musical patterns in a piece of music and we cannot always have $\gamma_P(X) = X$. However, to ensure this equality, we can consider the discovery of each pattern P_i separately and restrict X to $\gamma_{P_i}(X)$. This case will be covered in Section 3.5.

- X is bounded: In our analysis, the case where X is not bounded never happens because X represents a piece of music which is by definition bounded. There are not an infinite numbers of notes, nor an infinite value for duration, pitch or onset. Therefore, this assumption is necessary to ensure that Serra's proposition is true but is not a restriction for the musical applications.

3.2.4 Learning the Structuring Elements

Remark (Notations). In the following, we arbitrarily choose the first coordinate of $x \in E$ as the temporal component of x, we define $[\![a, b]\!]$ as the set of integers from a to b included, and $|A|$ the cardinality of the set A.

The objective of algorithms that discover patterns to describe musical data is to obtain a family $\mathcal{F}$ of structuring elements that verifies $\gamma_{\mathcal{F}}(X) \approx X$, which can also be expressed as follows $|\gamma_{\mathcal{F}}(X)| \approx |X|$. In this chapter, we propose the novelty of having two methods for learning the elements of $\mathcal{F}$: one can learn either the musical pattern P or its onsets O. Using Theorem 3.2, it is enough to identify only one of the two to obtain the other one (if P is full in X which is usually the case in music). In each case, it is necessary to proceed differently, but the main idea is to use X to learn the structuring element. The following lemma indicates that musical patterns or the onsets must be composed of sub-patterns in X, i.e. if a sub-pattern P' does not appear in X then it cannot be contained in musical patterns or the onsets.

Lemma 3.4: $\gamma_{P\cup P'}(X) = \emptyset$

Let $P, P', X \in \mathcal{P}(E)$. If $\gamma_P(X) = \emptyset$ then $\gamma_{P\cup P'}(X) = \emptyset$.

Proof of Lemma 3.4. This is due to the decreasingness of the opening with respect to the structuring element, i.e: $P \subseteq P \cup P' \Rightarrow \gamma_{P\cup P'}(X) \subseteq \gamma_P(X) = \emptyset$. □

From the previous lemma, we can deduce that the points of the structuring elements must belong to the set:

$$\{x_2 - x_1 \mid x_1, x_2 \in X \wedge x_1 \leq x_2\},$$

where $x_1 \leq x_2$ means that the lexicographic order of x_1 is lower than that of x_2. In other words, we can restrict ourselves to this set, rather than trying to cover the whole E set to discover structuring elements. This observation has already been made by Meredith et al. with the the SIA algorithm to optimize the discovery of musical patterns [Meredith 2002a], as described in Section 2.2.1. However, we propose here to give a musical meaning to the discovered patterns, distinguishing the role of pattern P and onsets O. Because many elements of this set are not musically interesting, additional conditions have to be added to find the appropriate structuring elements. Depending on whether we want to learn P or O, we then add different constraints.

3.2.4.1 Learning the Musical Patterns P

To reduce the possibilities to be tested to discover musical patterns P, we need to add more constraints that remove false candidates. For example, Collins proposed the SIAR algorithm, which discovers patterns whose points are not distant from the lexicographic order [Collins 2011]. This limits computations by discovering a subset of the patterns discovered by the SIA algorithm, as explained in Section 2.2.4. However, the possibilities remains very large and we propose another approach to discover musical patterns using the onsets.

3.2.4.2 Learning the Onsets O

One of the most important results of this article is the possibility of discovering musical patterns from the onsets. Under the assumptions mentioned in Theorem 3.2, it is possible to discover the musical pattern from the onsets with the morphological erosion. The major usefulness of this result arises from the algorithmic complexity reduction: it might be faster to discover the onsets than the patterns because there are far fewer possible choices.

In order to discover *exact repetitive patterns*, it is possible to consider structuring elements that represent time-periodic onsets. For example, if we are looking for

musical patterns P that repeat in time at regular intervals of length $L > 0$, the structuring element O associated with the onsets is:

$$\{(iL, 0, \ldots, 0) \mid i \in [\![0, m-1]\!]\}$$

In this case, we are looking for at least m exact repetitions of the musical pattern P. As a result, only one parameter L is required to discover patterns that repeat exactly at regular intervals. However, in the case where the patterns are not repeated exactly with a time translation, they can also be discovered with *transpositions*, which means that a pitch translation is also allowed. For example, we can first restrict ourselves to an up or down octave transposition, i.e. a pitch translation in $[\![-12, 12]\!]$. Representing the pitch by the second coordinate, the onsets O are part of the set:

$$\{(iL, p_i, 0, \ldots, 0) \mid i \in [\![0, m-1]\!] \wedge p_i \in [\![-12, 12]\!]\}$$

We can search for the values of L and p_i that are in the set defined in Section 3.2.4, choosing those that maximize the number of notes after an opening by these structuring elements. Therefore, only a few parameters L and p_i are required to discover patterns that repeat at regular intervals with transpositions, where each parameter has a limited number of values.

3.2.5 Links With Point-Set Algorithms to Discover Musical Patterns

The concept of onsets of a musical pattern has already been considered in point-set algorithms for discovering musical patterns with the SIATEC algorithm [Meredith 2001, Meredith 2002a, Meredith 2006]. In particular, we described in Section 2.2.2 that this algorithm produces the largest translatable patterns and their onsets. This is named differently, because what we call "onsets" here is referred to as "the set of translators" in the SIATEC algorithm. We have chosen the term "onsets" rather than "set of translators", as this facilitates understanding and interpretation of the discovered patterns by differentiating the role of patterns and onsets in a musical context. Therefore, although the discovery of pairs of musical patterns and onsets is not new, the mathematical results we propose are. To our knowledge, it has never been proven that it is possible to discover musical patterns from their onsets (or set of translators using the previous terminology), even though Collins had already discussed the idea of discovering a pattern from its conjugate (pp. 281-282) [Collins 2011]. Moreover, this approach generalizes the one developed by Meredith et al. with algorithms of the SIA family that discover MTP patterns, since these patterns can be interpreted as patterns discovered by onsets of two points x_1, x_2, because: $MTP(x_2 - x_2, X) = \varepsilon_{\{x_1,x_2\}}(X)$. This is why we propose to discover patterns using additional points $x_1, \ldots, x_m$ using $\varepsilon_{\{x_1,\ldots,x_m\}}(X)$ which is not possible with the MTP operation.

3.3 MTEC Conjugate Pair of Pattern/Onsets

In this section, we focus on pairs of musical patterns and their onsets which can be deduced from each other. We propose a theorem that proves the existence of such pairs in musical data.

3.3.1 Definition and Existence of an MTEC Conjugate Pair

In the following, we define the musical patterns P and their onsets O which verify the equality $P = \varepsilon_O(X)$ as *MTEC conjugate patterns*, and the pair (P, O) is a *MTEC conjugate pair*.

Definition 3.4: MTEC conjugate pair

Let $P, O, X \in \mathcal{P}(E)$. The pair (P, O) is a **MTEC conjugate pair** if it satifies:

$$P = \varepsilon_O(X) \quad \text{and} \quad O = \varepsilon_P(X)$$

Since the equation $P = \varepsilon_O(X)$ is not always true, it is in principle not always possible to discover an MTEC conjugate pair (P, O) of a musical pattern and its onsets. Therefore, given a musical pattern P, it cannot always be obtained from its onsets O with an erosion. This can be confirmed because several different patterns can have the same onsets. But the erosion of these onsets can only be equal to one of these patterns at most. With this in mind, Theorem 3.5 assures that given a pattern P, there exists an MTEC conjugate pair (P', O) such that the onsets of P' are the same as the onsets of P. Also, Lemma 3.1 ensures that: $P \subseteq P'$. Therefore, P' can be seen as P which has been enriched with the missing notes to obtain an MTEC conjugate pair.

Theorem 3.5: $O = \varepsilon_{P'}(X)$

Let $P, O, X \in \mathcal{P}(E)$, such that $O = \varepsilon_P(X)$ and $O \neq \emptyset$. By defining $P' = \varepsilon_O(X)$, we have:

$$O = \varepsilon_{P'}(X)$$

Proof of Theorem 3.5. We prove the lemma with two inclusions.
$\boxed{\subseteq}$ The first inclusion comes from Lemma 3.1 by exchanging the role of P and O by O and P'.
$\boxed{\supseteq}$ Using Lemma 3.1 we obtain: $P \subseteq \varepsilon_O(X)$. Because the erosion is decreasing

according to the structuring element we have:

$$P \subseteq \varepsilon_O(X) \Rightarrow \varepsilon_{\varepsilon_O(X)}(X) \subseteq \varepsilon_P(X)$$

Since $O = \varepsilon_P(X)$ and $P' = \varepsilon_O(X)$ by definition, this implies $\varepsilon_{P'}(X) \subseteq O$ which proves the other inclusion. □

Therefore, Theorem 3.5 can be reformulated with Corollary 3.6, which states that given the onsets O from a pattern P, there always exists a unique pattern P' containing P such that P' and O are MTEC conjugate patterns.

Corollary 3.6: P' and O are MTEC conjugate patterns

Let $P, O, X \in \mathcal{P}(E)$, such that $O = \varepsilon_P(X)$ and $O \neq \emptyset$. We have:

$$\exists! P' \in \mathcal{P}(E) : P \subseteq P' \wedge P' \text{ and } O \text{ are MTEC conjugate patterns}$$

Proof of Corollary 3.6. Let $P' = \varepsilon_{\varepsilon_P(X)}(X)$, we have $P \subseteq P'$ (Lemma 3.1), P' and O are MTEC conjugate patterns (Theorem 3.5), and uniqueness results from O having at most one MTEC conjugate. □

3.3.2 Musical Interpretations of an MTEC Conjugate Pair

In the example of Figure 3.2, we have seen that (P, O) is not an MTEC conjugate pair because $P \neq \varepsilon_O(X)$. Yet, according to Theorem 3.5, by defining $P' = \varepsilon_O(X)$, we obtain an MTEC conjugate pair with (P', O) where O is the onsets of P and P' is the pattern P which has been completed by the missing notes. This is illustrated in Figure 3.5 where the morphological erosion yields the MTEC conjugate pair (P', O) from P, and we can confirm that: $P \subseteq P'$.

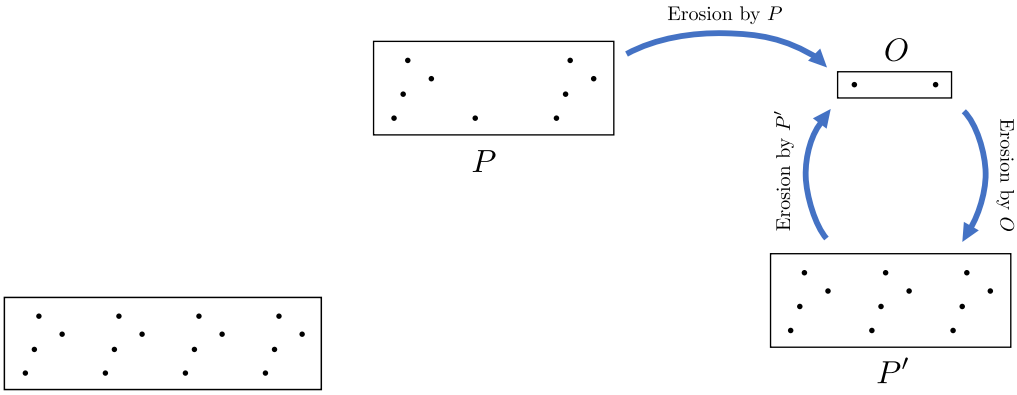
Remark (Relation between the pattern P and its onsets O). Under the assumptions of Theorem 3.2, we can switch from P to O with erosion:

$$P \underset{\varepsilon_O(X)}{\overset{\varepsilon_P(X)}{\rightleftarrows}} O$$

In this case, it is sufficient to know P (resp. O) to determine O (resp. P).

However, when the assumptions are not met, Theorem 3.5 states that it is enough to apply an erosion to get into a pattern/onsets cycle:

$$P \xrightarrow{\varepsilon_P(X)} O \underset{\varepsilon_{P'}(X)}{\overset{\varepsilon_O(X)}{\rightleftarrows}} P'$$



(a) X (b) Relations between P, O and P'

Figure 3.5: Using the example of Figure 3.2, (P, O) is not an MTEC conjugate pair because $P \neq \varepsilon_O(X)$. However, the morphological erosion by O provides the pattern P', by adding notes to P, such that (P', O) is an MTEC conjugate pair because $P' = \varepsilon_O(X)$ and $\varepsilon_{P'}(X) = O$.

Therefore, in any case, we can always obtain an MTEC conjugate pair of a pattern and its onsets from a musical pattern P. If it respects some properties, such as those of Theorem 3.2, this pair is trivial because it consists of the pattern and its onsets. Otherwise, we obtain the MTEC conjugate pair by completing the musical pattern with additional notes with a morphological erosion of the musical data as stated in Theorem 3.5.

3.3.3 Notes Covered With the Opening of an MTEC Conjugate Pair

The following lemma states that if we have an MTEC conjugate pair (P, O), then the notes covered with an opening by P are the same as the ones covered with an opening by O. This lemma is very useful in our situation because it demonstrates that the choice of P or O does not have any importance when we are interested in the notes covered with an opening.

Lemma 3.7: $\gamma_P(X) = \gamma_O(X)$

Let $P, O, X \in \mathcal{P}(E)$, such that $P = \varepsilon_O(X)$ and $O = \varepsilon_P(X)$. We have:

$$\gamma_P(X) = \gamma_O(X)$$

Proof of Lemma 3.7. Under the assumptions of the lemma, we directly prove the equality: $\gamma_P(X) = \delta_P(\varepsilon_P(X)) = \delta_P(O) = \delta_O(P) = \delta_O(\varepsilon_O(X)) = \gamma_O(X)$. □

This can be visualized in Figure 3.1, where the set of notes covered by the pattern opening is the same as the set of notes covered by the onsets opening. Therefore, Lemma 3.7 confirms the idea that it can be sufficient to focus on the onsets in the task of discovering musical patterns. In particular, it demonstrates that for algorithms that focus on pattern discovery to describe musical data, i.e. discover $\mathcal{F}$ such that $\gamma_{\mathcal{F}}(X) \approx X$, that $\mathcal{F}$ can be composed of onsets only without changing the set of notes covered by the expected result of the algorithm.

3.3.4 Links With Point-Set Algorithms to Discover Musical Patterns

Collins and Meredith have also proposed a definition of conjugate patterns [Collins 2011, Collins 2013, Meredith 2016]. It may seem different, but there is a resemblance between this definition and the one introduced in this chapter.

Let $P \in \mathcal{P}(E)$, and $TEC(P, X) = \{P_1, \ldots, P_m\}$ with $P_i = \{p_{i,1}, \ldots, p_{i,l}\}$ (see Section 2.2.2 for the TEC definition). Collins and Meredith started by defining the *conjugacy array* $J_{P,X}$, where each row of $J_{P,X}$ is an element of $TEC(P, X)$, formally:

$$J_{P,X} = \begin{pmatrix} p_{1,1} & \ldots & p_{1,l} \\ \vdots & & \vdots \\ p_{m,1} & \ldots & p_{m,l} \end{pmatrix} \begin{matrix} \leftarrow P_1 \\ \vdots \\ \leftarrow P_m \end{matrix}$$

By setting $O = \{p_{1,1}, \ldots, p_{m,1}\}$, the columns of $J_{P,X}$ constitute the elements of $TEC(O, X)$. Collins and Meredith define P and O as *conjugate patterns*, and added that $TEC(P, X)$ and $TEC(O, X)$ are *conjugate TECs*.

We demonstrate here that this is related to the definition introduced in this chapter. Using Lemma 2.2, we have: $TEC(P, X) = \{P_{t_1}, \ldots, P_{t_m}\}$ with $\varepsilon_P(X) = \{t_1, \ldots, t_m\}$. Thus, by setting $P = \{p_1, \ldots, p_l\}$, we have:

$$J_{P,X} = \begin{pmatrix} p_1 + t_1 & \ldots & p_l + t_1 \\ \vdots & & \vdots \\ p_1 + t_m & \ldots & p_l + t_m \end{pmatrix}$$

Therefore, the columns of $J_{P,X}$ are $\varepsilon_P(X)_{p_1}, \ldots, \varepsilon_P(X)_{p_l}$, which reveals that conjugate patterns (in the sense of Collins and Meredith) are linked by erosion. Their definition can be expressed using morphological terms:

$$P \text{ is conjugate with } O \iff \exists t \in E : O_t = \varepsilon_P(X)$$

However, this relation is not symmetric (P is conjugate with O does not imply that O is conjugate with P) and is not unique (there are several conjugates of P), so we have opted to revise this definition. To ensure that a pattern has at most

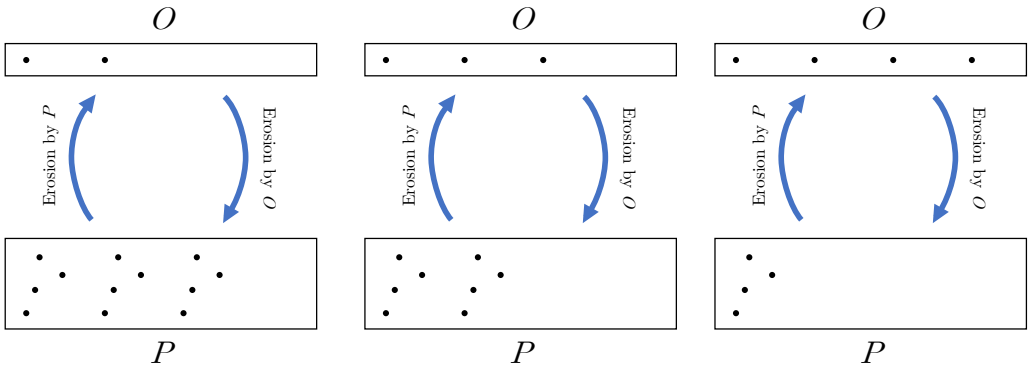
one conjugate, we impose that t is zero (i.e. $O = \varepsilon_P(X)$). Moreover, to have a symmetrical relation (P is conjugate with O implies O is conjugate with P), we added the condition that P can also be obtained by O with erosion (i.e. $P = \varepsilon_O(X)$). This explains Definition 3.4 used in this chapter for MTEC conjugate patterns.

3.4 Non-Redundant Pair of Pattern/Onsets

When we have musical data X, there are generally several MTEC conjugate pairs (P, O), i.e. which satisfy: $P = \varepsilon_O(X)$ and $O = \varepsilon_P(X)$. In this section, we define a criterion to determine which pair best represents the musical notions of musical patterns and onsets for P and O.

3.4.1 Definition of a Non-Redundant Pair

Figure 3.6 illustrates the three MTEC conjugate pairs (P, O) of a musical pattern P and its onsets O for X illustrated in Figure 3.5(a). In this case, it is not possible to find other MTEC conjugate pairs (P, O), with P or O different from a singleton. We might wonder which pair best represents the notion of musical patterns and onsets? To solve this problem, we want the musical pattern to be non-periodic to condense the information. Thus, the pair that best represents the musical pattern and its onsets is the one represented in Figure 3.6(c). In the other pairs, the pattern P is periodic and the information is redundant.



(a) $|P| \times |O| = 12 \times 2 = 24$ (b) $|P| \times |O| = 8 \times 3 = 24 = 24$ (c) $|P| \times |O| = 4 \times 4 = 16$

Figure 3.6: The three different MTEC conjugate pairs (P, O), i.e. with the property $P = \varepsilon_O(X)$ and $O = \varepsilon_P(X)$, of the set X represented in Figure 3.5(a). The (c) pair is non-redundant because it satisfies $|X| = |P| \times |O|$.

We use the following inequality to quantify this redundancy:

$$|\gamma_P(X)| = |P \oplus O| \leq |P| \times |O|,$$

where the equality ensures that each note $x \in X$ is uniquely expressed as $x = p + o$ with $p \in P$ and $o \in O$. In the case where the equality is not satisfied, there is at

least a note $x \in X$ such that $x = p + o = p' + o'$ with $p, p' \in P$, $o, o' \in O$ and $p \neq p', o \neq o'$, which means that the information is redundant. Therefore, we define (P, O) as *non-redundant* when the equality is met.

Definition 3.5: Non-redundant pair

Let $P, O, X \in \mathcal{P}(E)$, where (P, O) is an MTEC conjugate pair. The pair (P, O) is **non-redundant** if it satisfies:

$$|\gamma_P(X)| = |P| \times |O|$$

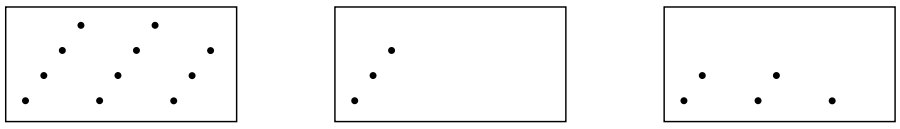
Note that from Lemma 3.7, we know that $\gamma_P(X) = \gamma_O(X)$ because (P, O) is an MTEC conjugate pair. Consequently, the choice of P or O has no importance in Definition 3.5 and we could change $\gamma_P(X)$ by $\gamma_O(X)$.

For example, in Figure 3.6, we have $|\gamma_P(X)| = |X| = 16$ for the three pairs. When we compute the cardinalities, we obtain $|P| \times |O| = 12 \times 2 = 24$ for the (a) pair, $|P| \times |O| = 8 \times 3 = 24$ for the (b) pair and $|P| \times |O| = 4 \times 4 = 16$ for the (c) pair. This allows us to conclude that the (c) pair is the one that best represents the musical pattern and its onsets.

3.4.2 Existence and Uniqueness of Non-Redundant Pairs

If we assume that there are several MTEC conjugate pairs (P, O) for given musical data X, we can wonder whether there is always exactly one non-redundant pair. In other words, is there existence and uniqueness of a non-redundant pair?

With respect to the existence, the set X, illustrated in Figure 3.7(a), includes several MTEC conjugate pairs like the one illustrated in Figure 3.7(b) and Figure 3.7(c). However, there is no non-redundant pair because $|X| = 11$ is a prime number. So X cannot be decomposed into a dilation of a musical pattern and its onsets without redundancy. This example demonstrates that there is not always a non-redundant pair.

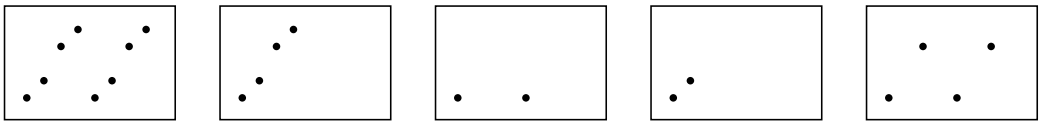


(a) X (b) P (c) O

Figure 3.7: Example of a set X where we cannot find a pair (P, O) such that $|X| = |P| \times |O|$. Here, $|X| = 11$ and with the (b)/(c) pair, we have $|P| \times |O| = 3 \times 5 = 15$.

With respect to uniqueness, in Figure 3.8, two different pairs are non-redundant,

which shows that uniqueness is not always true either. Therefore, there is not always existence and uniqueness of a non-redundant pair. However, this does not prevent the practical use of the proposed approach.



(a) X (b) P (c) O (d) P' (e) O'

Figure 3.8: Example of a set X where there are multiple pairs (P, O) such that $|X| = |P| \times |O|$. Here we have $|X| = 8$, $|P| \times |O| = 4 \times 2$ and $|P'| \times |O'| = 2 \times 4$.

3.4.3 Links With Point-Set Algorithms to Discover Musical Patterns

To select the pair that best represents the concept of a musical pattern and its onsets, we used the non-redundancy criterion presented in Definition 3.5. However, other criteria have already been mentioned in point-set algorithms for selecting salient pairs from the many discovered ones. In particular, Meredith et al. have used the "compression ratio" to select the best pair (P, O) (which is called a TEC in their method) in the COSIATEC algorithm [Meredith 2002a, Meredith 2006, Meredith 2013] presented in Section 2.2.5, and which consists in maximizing the quantity:

$$\frac{|\gamma_P(X)|}{|P| + |O| - 1} \in [0, \infty[$$

Because it has no upper bound, to select the pair with the highest compression ratio, we need to compute the value of all discovered pairs and choose the one with the maximum value. In addition, in this formula, there is no guarantee that there are no intersections between occurrences of the pattern P in the musical data X. We therefore proposed to maximize the non-redundancy instead of the compression ratio with the following formula:

$$\frac{|\gamma_P(X)|}{|P| \times |O|} \in [0, 1]$$

This formula does not compute the compression ratio, but it does have a few advantages with respect to the method we use to discover musical patterns. If the value 1 is reached, this means that $|\gamma_P(X)| = |P| \times |O|$, i.e. the occurrences of P do not intersect in X. It is therefore not necessary to compute the redundacy of all possible pairs to obtain non-redondant pairs, which are considered salient. This is why we have chosen to restrict ourselves to discover MTEC conjugate pairs that are non-redundant to describe the musical data.

3.4.4 Existence of a Non-Redundant MTEC Conjugate Pair With a Non-Periodic Pattern

We consider in this section the particular case of a non-periodic pattern P repeated at a period L. This situation, although it seems simple, is very common due to the importance of repetition in music [Janssen 2013]. In this case, Theorem 3.8 assures that because of its non-periodicity, the pattern P does not appear elsewhere than at distance L. Consequently, this theorem proves the existence of a non-redundant MTEC conjugate pair with P and the onsets $\{kL \mid k \in [\![0, N-1]\!]\}$, that can be interpreted as points separated by a distance L.

Theorem 3.8: MTEC conjugate pair with a non-periodic pattern

Let $X = P \oplus O$ with $P, O \in \mathcal{P}(E)$ such that:

- $L = (t_L, 0, \ldots, 0) \in E$ with $t_L > 0$,
- P is of temporal length less than t_L,
- $O = \{kL = (kt_L, 0, \ldots, 0) \mid k \in [\![0, N-1]\!]\}$ with $N \in \mathbb{N}^*$,
- P is non-periodic in all directions (meaning $\nexists S \in \mathcal{P}(E)$ such that $S \subsetneq P$, S is not a singleton and $\gamma_S(P) = P$).

With these assumptions, we have:

(P, O) is a non-redundant MTEC conjugate pair

In the assumptions of this theorem, P is of temporal length less than t_L signifies that $\max_{x,y \in P} |t_x - t_y| < t_L$, where t_x and t_y are the temporal components of x and y.

Remark. Compared to Theorem 3.2, we do not have the condition $\varepsilon_P(X) = O$ in the assumptions. However, it is satisfied due to the non-periodicity of P.

This theorem reveals another application of the definitions introduced in this chapter. In particular, it demonstrates that the discovery of specific onsets can produce non-redundant MTEC conjugate pairs (P, O) where the pattern P is non-periodic. In the assumptions of the theorem, if the pattern P is temporally periodic, for example by considering $P \cup P_L$ instead of P and $O = \{2kL\}_k$ instead of $O = \{kL\}_k$ (because $P \cup P_L$ is of temporal length less than $2t_L$), the result is false and no MTEC conjugate pair is obtained, since $\varepsilon_{P \cup P_L}(X) \neq \{2kL\}_k$. Moreover, the MTEC conjugate pair $(P \cup P_L, \varepsilon_{P \cup P_L}(X))$ is not non-redundant, which demonstrates that the discovery of non-periodic patterns is related to the discovery of non-redundant MTEC conjugate pairs. Finally, the proof of Theorem 3.8 is in the Appendix A.

3.5 Applications of the Proposed Method to Symbolic Representations of Music

In this section, we apply the method proposed in this chapter to discover patterns in musical data. According to the GTTM [Lerdahl 1985], we can assume that the musical patterns are full because they do not intersect in music, or at worst they just share the first and last note. This allows us to use one of the main theorems of this chapter, Theorem 3.2, to first learn the onsets and then deduce the musical patterns with an erosion in order to obtain MTEC conjugate pairs of musical pattern and onsets. The learning process of the structuring element, here the onsets, is obtained by maximizing the number of notes after an opening and then selecting a non-redundant pair of musical pattern and its onsets as described in Section 3.4. We therefore demonstrate that musical patterns can be discovered from their onsets and show that a musical piece can be approximated as a union of dilations between patterns and onsets.

3.5.1 Symbolic Representations of Music to Apply Mathematical Morphology

The first thing to consider when applying mathematical morphology tools is how to represent musical data. As we described in Section 1.2, the musical data must belong to $E = \mathbb{R}^n$, $\mathbb{Z}^n$ or $\mathbb{Z}_n$ because this is the context of binary mathematical morphology. This is well adapted to symbolic representation of music because each note can be defined by an n-uplet as it has been done before by Meredith et al. to discover musical patterns [Meredith 2002a]. We propose here to work in $\mathbb{R}^3$, where each note is a unique point defined by (o, p, d), where O is the note onset (i.e. when the note appears in time), P is the pitch of the note and d its duration. It is also possible to add other dimensions such as velocity. However, this representation in $\mathbb{R}^3$ by the onset, the pitch and the duration of each note is a correct compromise to obtain a good description of a piece of music without being too costly from a computational point of view. Moreover, this representation ensures that the number of notes of a musical piece X is given by $|X|$. Previous works using mathematical morphology applied to music have represented musical data in $\mathbb{R}^2$ with a piano roll representation. This is the most intuitive one and therefore the one used in our illustrations. Karvonen et al. [Karvonen 2008, Karvonen 2010] have chosen to represent the duration in the piano roll. In this case, each note is no longer a single point but a segment of length proportional to the duration of the note. From a computational point of view, this increases the computation time considerably because $|X|$ is much larger than the number of notes in X, which is why we did not choose to follow this path. However, this idea would be particularly well suited for pattern discovery in music performance because it would allow for a temporal approximation when learning the structuring elements.

Therefore, in the rest of this chapter, the musical data belong to $\mathbb{R}^3$, where

each note of X is represented by (o, p, d), respectively its note onset, pitch and duration. Thereafter, the morphological operators are applied in $\mathbb{R}^3$. However, to avoid having figures in $\mathbb{R}^3$ that are difficult to interpret, we project the result in a piano roll representation to simplify understanding.

3.5.2 Discover the Patterns from the Onsets and Summarize the Musical Data With Morphological Operators

We apply here the proposed method to a musical example in order to discover the main musical patterns and to approximate the musical data with a union of dilations between patterns and onsets.

Regarding the musical data X, we choose here the guitar part of the musical piece *Deserted Dunes Welcome Weary Feet* of the band King Gizzard and the Lizard Wizard. This song is copyright free and the midi files are freely available[2], which makes it a good candidate for music information retrieval research. The musical example X is illustrated in Figure 3.9 using a piano roll representation.

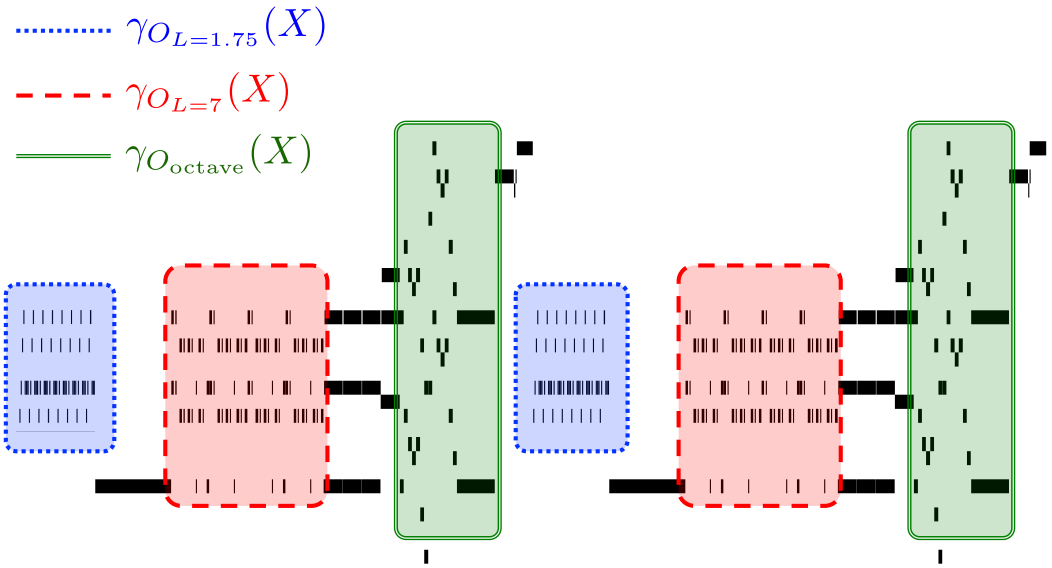


Figure 3.9: Representation of the musical data X from the midi file of the piece *Deserted Dunes Welcome Weary Feet* (guitar voice). X is composed of 366 notes, $\gamma_{O_{L=1.75}}(X)$, $\gamma_{O_{L=7}}(X)$ and $\gamma_{O_{\text{octave}}}(X)$ discover respectively 112, 144 and 60 notes of X. In any cases, the morphological openings can be decomposed as a dilation of a musical pattern by its onsets.

In this case, X is composed of 366 notes, i.e. $|X| = 366$, and we aim to discover a family of structuring elements $\mathcal{F}$ which represents musical patterns, with the criteria

[2]The whole album is available as midi files where the tracks are separated at: https://kglw8-bit.bandcamp.com/album/polygondwanaland-8-bit-w-tabs

set out in Section 3.4, such that $|\gamma_{\mathcal{F}}(X)|$ is as close as possible to $|X|$.

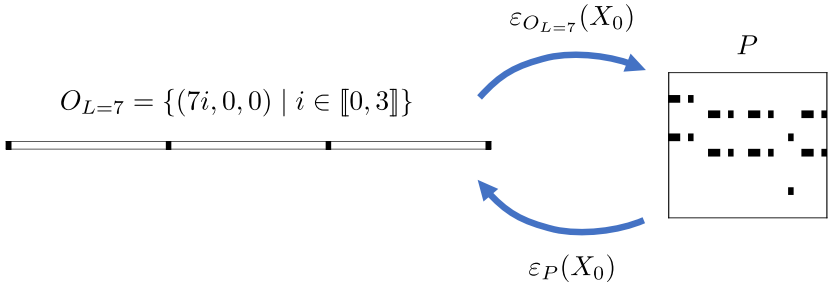
First, we can discover exact repetitions on a large scale, i.e. if sections repeat. To do this, we must look for structuring elements that preserve as many notes as possible after an opening of X in the form $A = \{(0,0,0),(L,0,0)\}$, with L large enough. Let L_{max} be the temporal length of X. We notice that the structuring element $A = \{(0,0,0),(L_{max}/2,0,0)\}$ satisfies: $\gamma_A(X) = X$. This implies that X can be divided into two identical parts with respect to time. Therefore, it is enough to study X only on its first half, i.e. it is enough to study X on $\varepsilon_A(X)$. Since $|\varepsilon_A(X)| = \frac{|X|}{2} = 183$, this allows the data to be reduced considerably. In the following, we define the first half of X as X_0, which is equal to $\varepsilon_A(X)$, to simplify notations and we learn the structuring elements using X_0. Similar reasoning can be used to discover repeating sections, even if X is not divided exactly into two equal parts. In the following, we set the temporal axis so that 1.0 is the duration of a quarter note.

According to Section 3.2.4.2, we know the shape of the onsets O if we want to discover musical patterns P that repeat exactly or with transposition. We can then start by discovering the musical patterns P that repeat exactly 4 times in X by looking for the structuring elements in the form:

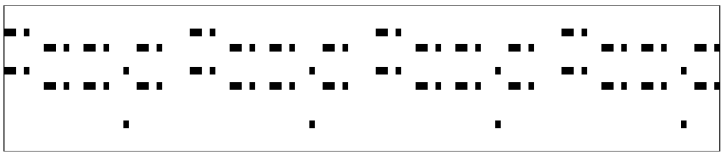
$$O_L = \{(0,0,0),(L,0,0),(2L,0,0),(3L,0,0)\},$$

where L belongs to $\mathbb{R}$ because the musical data X belong to $\mathbb{R}^3$, so L is not necessarily an integer. By varying L, we can learn the onsets O_L that maximize $|\gamma_{O_L}(X)|$. From the previous paragraph, it is equivalent to maximizing $|\gamma_{O_L}(X_0)|$, i.e. just processing the first half of X. Thus, the value of L that maximizes $|\gamma_{O_L}(X_0)|$ is reached for $L = 7$ with $|\gamma_{O_{L=7}}(X_0)| = 72$. This means that $72 \times 2 = 144$ notes, among the 366 of X, are part of a musical pattern that is repeated with a period of $L = 7$ (the factor 2 in 72×2 comes from the fact that we used X_0 which is the first half of X). According to Theorem 3.2, we can then use an erosion of X_0 by the structuring element $O_{L=7}$ to discover the corresponding musical pattern P. The onsets $O_{L=7}$ and its associate musical pattern $P = \varepsilon_{O_{L=7}}(X_0)$ are represented in Figure 3.10(a) using a piano roll representation. Because $|P| \times |O_{L=7}| = |\gamma_{O_{L=7}}(X_0)|$ we can conclude that $(P, O_{L=7})$ is a non-redundant pair, based on Section 3.4, in the sense that it best interprets the notions of musical pattern and its onsets without redundancy of information. The set $\gamma_{O_{L=7}}(X)$ is represented in red and framed by long dotted lines in Figure 3.9. We have just discovered the first structuring element $O_{L=7}$ of the family $\mathcal{F}$ which allows to describe X as closely as possible. To identify other structuring elements using the same process, the set $\gamma_{O_{L=7}}(X_0)$ is removed from X_0 with the top hat transform operation, which is another morphological operator that removes the results of the opening from the initial set: $T_{O_{L=7}}(X_0) = X_0 \backslash \gamma_{O_{L=7}}(X_0)$, and the discovery of other structuring elements is done with $X_1 = T_{O_{L=7}}(X_0)$.

Continuing the process, the value of L that maximizes $|\gamma_{O_L}(X_1)|$ is reached for $L = 1.75$. In this case, we have $|\gamma_{O_{L=1.75}}(X_1)| = 56$, which means that $56 \times 2 = 112$ notes of X are part of a pattern that repeats with a period of $L = 1.75$. The onsets



(a) $(P, O_{L=7})$ pair where the pattern P is discovered using the erosion by $O_{L=7}$

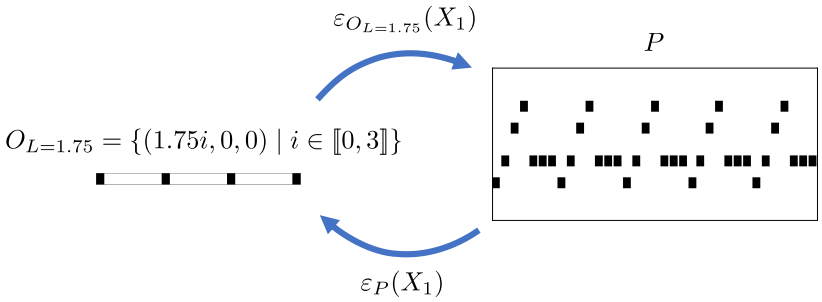


(b) $\gamma_{O_{L=7}}(X_0) = O_{L=7} \oplus P$

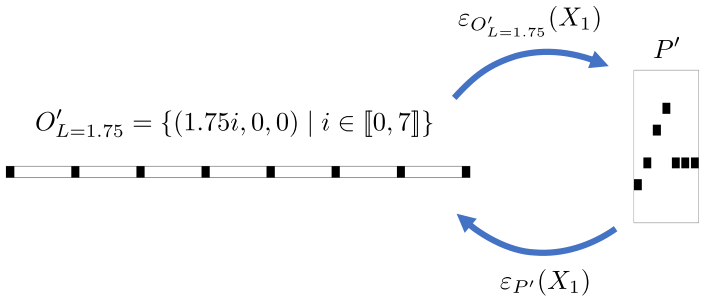
Figure 3.10: Discovery of repeated patterns by learning the onsets with the exact repetition constraint when $L = 7$. Because $|\gamma_{O_{L=7}}(X_0)| = |P| \times |O_{L=7}|$, we conclude that $(P, O_{L=7})$ is a non-redundant pair.

$O_{L=1.75}$ and its corresponding musical pattern P, obtained by the erosion of X_1 by $O_{L=1.75}$, are represented in Figure 3.11(a) using a piano roll representation. However, since $|O_{L=1.75}| \times |P| = 140 \neq 56$, the pair $(P, O_{L=1.75})$ is not non-redundant according to Section 3.4. By looking for 8 but not 4 exact repetitions of a musical pattern with $O'_{L=1.75} = \{(1.75i, 0, 0) \mid i \in [\![0, 7]\!]\}$, we get a non-redundant pair because $|O'_{L=1.75}| \times |P'| = 56$, with $P' = \varepsilon_{O'_{L=1.75}}(X_1)$. This non-redundant pair $(P', O'_{L=1.75})$ is shown in Figure 3.11(b). Unlike P, it can be seen that P' is not periodic and therefore better corresponds to what we expect from a musical pattern. The result obtained with the morphological filter $\gamma_{O_{L=1.75}}(X)$ is shown in Figure 3.9 in blue and framed by small dotted lines. As before, we remove $\gamma_{O_{L=1.75}}(X_1)$ from X_1 by defining $X_2 = T_{O_{L=1.75}}(X_1)$ to discover the next structuring elements.

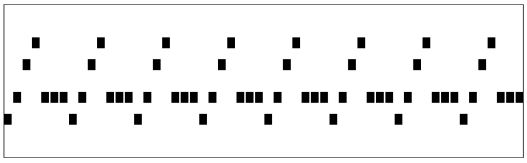
Then, we can also consider repetitions with respect to the pitch. For example, octave intervals can be discovered with the structuring element $O_{\text{octave}} = \{(0, 0, 0), (0, 12, 0)\}$. As explained in Section 1.3.3, the first coordinate of the elements of O_{octave} is 0 to discover notes that appear at the same time, and the last coordinate is 0 to discover notes that have the same duration. With this structuring element, we obtain $|\gamma_{O_{\text{octave}}}(X_2)| = 30$. Thus, at least $30 \times 2 = 60$ notes of X are part of an octave interval. By setting $P = \varepsilon_{O_{\text{octave}}}(X_2)$, we obtain the musical



(a) $(P, O_{L=1.75})$ pair where the pattern P is discovered using the erosion by $O_{L=1.75}$



(b) $(P', O'_{L=1.75})$ pair where the pattern P is discovered using the erosion by $O'_{L=1.75}$

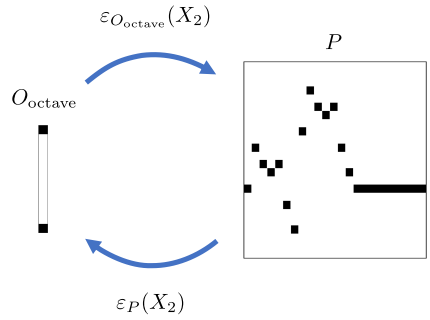


(c) $\gamma_{O_{L=1.75}}(X_1) = O_{L=1.75} \oplus P = O'_{L=1.75} \oplus P'$

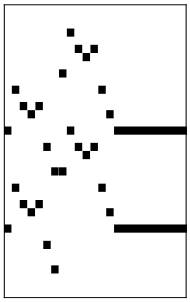
Figure 3.11: Discovery of repeated patterns by learning the onsets with the exact repetition constraint when $L = 1.75$. Because $|\gamma_{O_{L=1.75}}(X_1)| = |P'| \times |O'_{L=1.75}|$, we conclude that $(P', O'_{L=1.75})$ is a non-redundant pair.

pattern that is repeated at the octave. Since $|O_{\text{octave}}| \times |P| = 30$, we can deduce that (O_{octave}, P) is non-redundant. The two structuring elements O_{octave} and P are represented in Figure 3.12(a). Moreover, the set $\gamma_{O_{\text{octave}}}(X)$ is shown in Figure 3.9 in green and framed with double lines.

Finally, by setting $\mathcal{F} = \{O_{L=7}, O'_{L=1.75}, O_{\text{octave}}\}$, the set $\gamma_{\mathcal{F}}(X)$ is a fairly good approximation of X in the sense that $|\gamma_{\mathcal{F}}(X)| = 316$. Since $|T_{\mathcal{F}}(X)| = 50$, there are only 50 notes out of 366 that have not been discovered by $\gamma_{\mathcal{F}}(X)$. We can then



(a) (P, O_{octave}) pair where the pattern P is discovered using the erosion by O_{octave}



(b) $\gamma_{O_{\text{octave}}}(X_2) = O_{\text{octave}} \oplus P$

Figure 3.12: Discovery of repeated patterns by learning the onsets with octave transposition constraint. Because $|\gamma_{O_{\text{octave}}}(X_2)| = |P| \times |O_{\text{octave}}|$, we conclude that (P, O_{octave}) is a non-redundant pair.

describe X by morphological operations:

$$X \approx \bigcup_{O \in \mathcal{F}} O \oplus \varepsilon_O(X)$$

3.5.3 Other Applications: Using the Method to Discover Hidden Patterns in Music

The method developed in this chapter can also be used to discover patterns that are not full in X. In some cases, this may allow discovering patterns that were not clearly discernible when listening. For example, let X be the main musical theme of *Piano Phase* by Steve Reich, represented in Figure 3.13(a). In this case X is non-periodic, in the sense that $\nexists S \subsetneq X$ such that S is not a singleton and $\gamma_S(X) = X$. However, the method based on mathematical morphology proposed in this chapter allows us to better understand this musical theme.

By trying to learn a family of structuring elements that allows describing X with morphological operations, we notice that the pairs of musical patterns and onsets (P, O) and (P', O') represented in Figure 3.13 satisfy:

$$X = \gamma_P(X) \cup \gamma_{P'}(X),$$

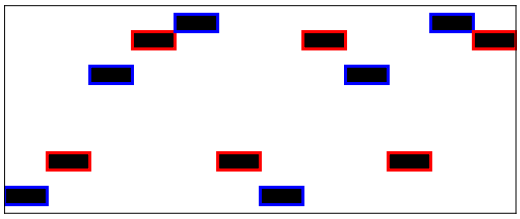
with the following links between the musical patterns and their onsets, meaning that they are both MTEC conjugate pairs:

$$\varepsilon_P(X) = O, \quad \varepsilon_O(X) = P, \quad \varepsilon_{P'}(X) = O', \quad \varepsilon_{O'}(X) = P'$$

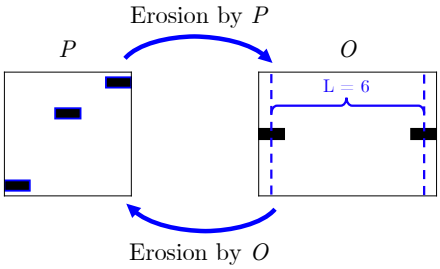
It is then possible to represent the theme of *Piano Phase* by the following equation:

$$X = (P \oplus O) \cup (P' \oplus O')$$

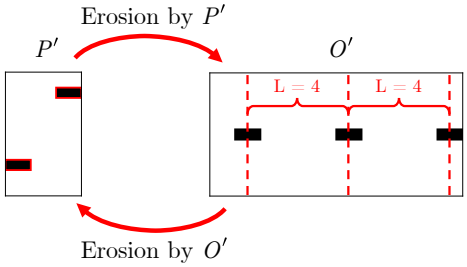
Thus, despite the lack of obvious patterns in X, it is still possible to discover hidden patterns in X using mathematical morphology to simplify its description.



(a) X



(b) (P, O) pair where $\varepsilon_P(X) = O$ and $\varepsilon_O(X) = P$



(c) (P', O') pair where $\varepsilon_{P'}(X) = O'$ and $\varepsilon_{O'}(X) = P'$

Figure 3.13: Musical theme of *Piano Phase* (a) composed of two MTEC conjugate pairs (P, O) and (P', O') of musical patterns and their onsets. The musical theme can be describe with morphological operations: $X = (P \oplus O) \cup (P' \oplus O')$.

3.6 Summary, Future Work and Conclusion

In this section, we summarize the principal definitions and results introduced in this chapter for discovering musical patterns using mathematical morphology. We then propose some directions for future work to extend this approach and we finally conclude this chapter.

3.6.1 Summary

In this chapter, we have proposed an original approach based on mathematical morphology for discovering musical patterns using a multidimensional representation of music. To do this, we have introduced several definitions which are summarized in Table 3.1. The fundamental definition provided in this chapter is the definition of onsets using morphological erosion, which allows us to distinguish the role of patterns and onsets for the discovery of musical patterns. We then defined a pattern as full as a convex set in the musical data (i.e. without holes), which makes the link with the topology present in image processing, where morphological operators are mostly applied. This definition makes it possible to apply some morphological results to musical data, and is necessary because objects are fundamentally different (objects are connected sets in image processing, while those associated with symbolic representations of music are sparse and therefore never connected). In particular,

we proved that with this definition, the problem of discovering a musical pattern from its onsets is satisfied. Finally, we characterized the salient pairs of musical patterns and their onsets as MTEC conjugate and non-redundant, which allows us to discover relevant pairs without having to discover and analyze all the pairs in the musical data.

Table 3.1: Summary of the principal definitions introduced in this chapter for discovering musical patterns using mathematical morphology, where $P, O, X \in \mathcal{P}(E)$.

Definitions	**Description using morphology**
Onsets of P in X	$O = \varepsilon_P(X)$
Discovering pattern from its onsets	$P = \varepsilon_O(X)$
P is **full** in X	$\forall t \in E, P_t \subseteq X \Rightarrow \mathrm{Ch}(P_t) \cap X = P_t$
MTEC conjugate pair (P, O) in X	$P = \varepsilon_O(X)$ and $O = \varepsilon_P(X)$
Non-redundant pair (P, O) in X	$\lvert\gamma_P(X)\rvert = \lvert P\rvert \times \lvert O\rvert$

The approach proposed in this chapter produces several mathematical results which are summarized in Table 3.2. The majority of these results can be interpreted musically and provide a better understanding of the task of discovering musical patterns.

- Lemma 3.1: The onsets of the onsets of a pattern enlarge it.
- Theorem 3.2: A pattern that is full can be discovered from its onsets.
- Theorem 3.5: A pattern discovered with the erosion is always MTEC conjugate with its onsets.
- Corollary 3.6: There always exists a unique pattern that is MTEC conjugate with the onsets of a given pattern.
- Lemma 3.7: If MTEC conjugate, the notes covered by a pattern or its onsets are the same.
- Theorem 3.8: A non-periodic pattern repeated at regular intervals produces a non-redundant MTEC conjugate pair.

Table 3.2: Mains results presented in this chapter that introduce an original approach based on algebraic formalism for discovering musical patterns, where $P, X \in \mathcal{P}(E)$.

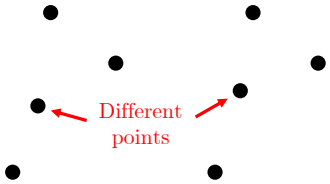
Name	Results expressed with mathematical morphology
Lemma 3.1	$P \subseteq \varepsilon_{\varepsilon_P(X)}(X)$
Theorem 3.2	P is full in $X \Rightarrow P = \varepsilon_{\varepsilon_P(X)}(X)$
Lemma 3.3	$\varepsilon_{\varepsilon_P(X)}(X) = \varphi_{\varepsilon_P(X)}(P)$
Theorem 3.5	$\varepsilon_P(X) = \varepsilon_{\varepsilon_{\varepsilon_P(X)}(X)}(X)$
Corollary 3.6	$\exists! P' : P'$ and $\varepsilon_P(X)$ are MTEC conjugate
Lemma 3.7	P and $\varepsilon_P(X)$ MTEC conjugate $\Rightarrow \gamma_P(X) = \gamma_{\varepsilon_P(X)}(X)$
Theorem 3.8	P non-periodic $\Rightarrow P$ and $\{kL\}_k$ MTEC conjugate non-redundant

3.6.2 Future Work

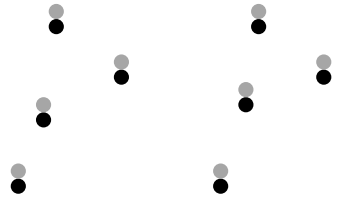
We propose here some directions for future work to develop the approach proposed in this chapter for discovering musical patterns from a multidimensional representation of music using mathematical morphology.

- **Discover non-repeating patterns:** Patterns discovered with the method developed in this chapter are patterns that repeat up to a translation, for example up to a harmonic or temporal translation. However, it is possible to adapt this method to discover patterns that repeat up to a translation, but with an approximation. For example, in Figure 3.14(a), a pattern is repeated with a variation on the second point. Musically, this can be interpreted as a note that is a semi-tone higher. To discover patterns with these variations, we can use the main idea of Karnoven et al. [Karvonen 2008, Karvonen 2010], which is to apply a morphological dilation to the musical data to match the patterns with an approximation. In our case, we want to add a harmonic tolerance. This can be done by dilating the musical data with $\{(0,0,0),(0,1,0)\}$ as structuring element in order to obtain an approximation of one semi-tone to discover the musical patterns (representing the data in $\mathbb{R}^3$, as explained in Section 3.5.1, with the pitch as second coordinate). The dilation of the musical

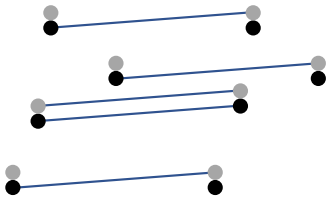
data in Figure 3.14(a) by this structuring element is shown in Figure 3.14(b), where the gray dots are those added after this operation. We can then use the method proposed in this chapter, i.e. apply morphological erosion by the onsets to discover the musical patterns, as illustrated in Figure 3.14(c). The result of the erosion, displayed in Figure 3.14(d), reveals the pattern and its variations. However, there is still work to be done to correctly identify the right method and the structuring elements to be applied according to the desired approximations with reasonable computing time. Finally, this method could also be used to discover patterns in performance, which can vary according to the expressivity of the performance, by using a dilation to obtain a temporal approximation.



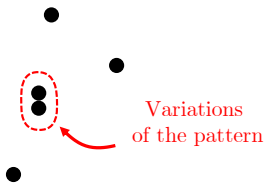
(a) Musical data composed of two slightly different patterns.



(b) Dilation of the musical data to obtain an approximation in the discovery of musical patterns.



(c) Erosion of the musical data that have been dilated.



(d) Discovery of the pattern and its variations from musical data.

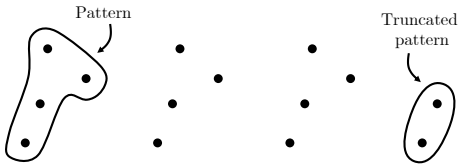
Figure 3.14: Dilation of musical data before applying the method proposed in this chapter to discover musical patterns with an approximation. This provides the variations of musical patterns.

- **Discover truncated patterns:** Another situation in which patterns do not repeat identically is when patterns are truncated in musical data. This technique is frequently used by specific musical artists [Pieslak 2007]. For example, in Figure 3.15(a), the musical data X is composed of a pattern P that is repeated three times and truncated the fourth time. In this case, the pattern P can not be simply discovered with just one erosion because there is no non-redundant MTEC conjugate pair for this musical data, specifically, it is not possible to apply Theorem 3.2 because $\gamma_P(X) \neq X$. However, morphological operators can still be used to reveal significant information. In particular, the

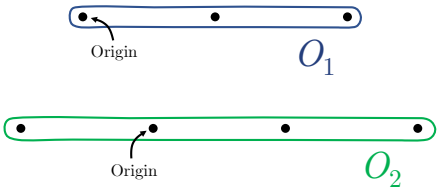
morphological erosion $\varepsilon_{O_2}(X)$ provides the truncated version of the pattern with O_2 composed of four points as illustrated in Figure 3.15(b). Moreover, the pattern P is discovered from two morphological erosions with:

$$P = \varepsilon_{O_1}(X) \setminus \varepsilon_{O_2}(X),$$

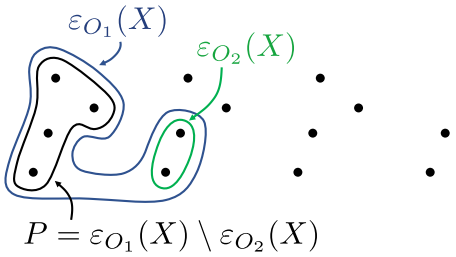
where O_1 contains one less point than O_2, as illustrated in Figure 3.15(c). Therefore, it may be interesting to develop this idea to identify the appropriate O_1 and O_2 and adapt the definition of non-redundant MTEC conjugate pair to this situation to discover patterns and their truncated version.



(a) Particular case where the last pattern repeat is truncated.



(b) The onsets that reveal the pattern and its truncated version.



(c) Erosion of musical data to discover the pattern from two erosions.

Figure 3.15: Illustration of musical data with a truncated pattern. In this case, the pattern can be discovered from two erosions.

- **Detection of the meter using discovered onsets:** We believe that the method developed in this chapter is relevant to the meter detection task, which is a field of research that consists in detecting meter from a symbolic representation of music [De Haas 2016, McLeod 2017]. Indeed, the discovery of the onsets produces important information about the meter. For example, we have seen in Figure 3.9 that the onsets are points separated by distances $L = 7$ and $L = 1.75$, which leads to a meter equal to $\frac{7}{4}$ or $\frac{7}{8}$ for this musical data (because $1.75 \times 4 = 7$). The choice of $\frac{7}{4}$ or $\frac{7}{8}$ can also be decided and adjusted according to the MTEC conjugate pattern of the onsets. The advantage of

this approach is that it requires no prior training, and can therefore be used for any type of meter, especially irregular meters where it is very difficult to obtain annotated databases for training. We therefore believe that it would be worthwhile to develop this idea to detect the meter from the discovered onsets and patterns. An original project would be to test it on a database containing a large number of irregular meters, such as the one set out in Section 3.5.2, which is not often addressed in the meter detection task.

- **Discover sections as a dilation of a pattern and its onsets for the segmentation task:** The discovery of pairs of patterns and their onsets can be used to describe musical sections with the morphological dilation, as explained in Section 3.2.3. For example, the three pairs discovered in Section 3.5.2 cover the three different main sections, as illustrated in Figure 3.9. This is because musical patterns and their onsets influence the segmentation of musical data [Cambouropoulos 2006]. We therefore expect that this approach for discovering patterns and onsets could also be used for the segmentation task. Because most approaches to obtain the segmentation from a symbolic representation of music use a string representation of music, it would be original to develop a new method based on a multidimensional representation of music. This was started by Meredith [Meredith 2016], and we believe that some of the ideas developed in this chapter (the discovery of non-redundant MTEC conjugate pairs of patterns and onsets) may provide a deeper understanding of discovering musical sections for the segmentation task using a multidimensional representation of music.

3.6.3 Conclusion

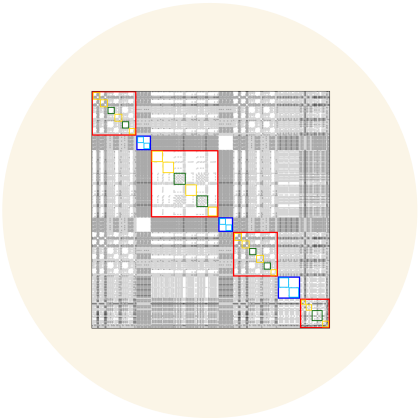
In this chapter, we have developed an original application of mathematical morphology to symbolic representations of music. Specifically, we focused on the discovery of musical patterns and demonstrated the relevance of basic morphological operators for this task. We proved several mathematical results that demonstrate the relevance of mathematical morphology for the discovery of musical patterns using a multidimensional representation of music. The main result is that it is possible to discover the musical pattern from its onsets with a morphological erosion if the patterns do not intersect, which is coherent in a musical context. This result is pertinent because it is usually simpler to discover the onsets than the musical pattern due to repetitions in the musical data. Therefore, the developed method is interesting because it starts from musical problems as a main motivation, i.e. the results and assumptions of the theorems are designed to be applied to musical data, and produces mathematical theoretical results that can also be applied to other types of discrete set data. Finally, we expect in future research that the use of mathematical morphology to symbolic representations of music will be a tool developed for both musical analysis and generation, while providing interesting and original mathematical results.

Part II

Musical Structure

Chapter 4

Homogeneity-Based Algorithm for Generating Hierarchical Segmentations of Music



Contents

This chapter deals with the computational analysis of musical structures by focusing on the use of mathematical morphology. In particular, the main contribution of this chapter is to demonstrate that morphological filters can be used to homogenize and detect regions in the self-distance matrix. Specifically, the opening operation has been successfully applied to reveal the blocks around the diagonal because it removes small details such as high local values and reduces all blocks around the diagonal to a zero value. Moreover, by varying the size of the morphological filter, it is possible to detect musical structures at different scales. A large opening filter identifies the main global parts of the piece, while a smaller one finds shorter musical sections. We discuss some examples that demonstrate the usefulness of this approach to detect the structures of a musical piece and its novelty within the field of symbolic music information research.

Section 4.1 explains the novelty and context of this method. Section 4.2 introduces the definitions of grayscale mathematical morphology that are useful in this chapter. Section 4.3 presents the self-distance matrix and describes a method for generating it from symbolic musical representations, proposing the chord contour as a generalization of the melodic contour. Section 4.4 describes how to use morphological filters in order to extract musical structures from the self-distance matrix, and illustrates how changing the size of the filters can be used to discover musical structures at multiple levels of granularity.

This chapter is an extended version of the article "Computational Analysis of Musical Structures based on Morphological Filters" [Lascabettes 2022a].

4.1 Introduction

In Part I, we have demonstrated that the theory of binary mathematical morphology is relevant to symbolic representations of music. In particular, in Chapter 1, we have described the relevance of basic morphological operators for music analysis and generation. In Chapter 2 and Chapter 3, we have proved that this theory provides a deeper understanding and generalization of previous works on musical pattern discovery. However, we have so far restricted the scope of our work to binary mathematical morphology. In this chapter, we propose an application of grayscale mathematical morphology to music data, which is a generalization of binary mathematical morphology to functions. This enables morphological filters to be applied to grayscale images where each pixel has a real value, unlike binary images. But grayscale mathematical morphology is algorithmically more complex, which is why it may not be used in exactly the same applications as in the previous chapters. Therefore, we propose to work on self-distance matrices. This representation is fundamental to the analysis of musical structures because blocks around the diagonal provide structural information on a musical piece.

The main contribution of this chapter is to propose a novel method, based on

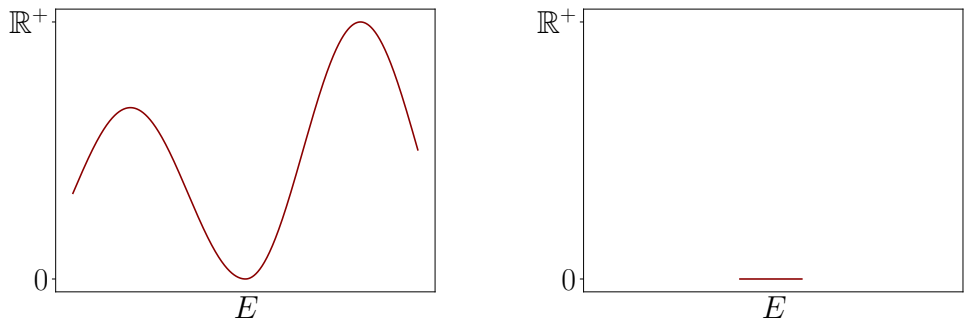
mathematical morphology, to extract hierarchical musical structures from the self-distance matrix. This method can be applied to any type of similarity matrix and to any type of data. In our case, the self-distance matrix is computed from symbolic music representations, using a generalization of melodic contour to chord sequences. The purpose of this method is to homogenize the different regions of the self-distance matrix in order to identify the musical structures. Two basic morphological operations, the erosion and dilation, have already been successfully used to detect the repeating patterns longer than a minimum length into a time-lag matrix (a similar representation as the self-distance matrix) [Lu 2004]. However, rather than identifying segments as in [Lu 2004], we demonstrate the usefulness of the morphological opening operation in order to identify blocks in the self-distance matrix. This operation eliminates small details, while flatter and homogeneous regions are obtained. In addition, it reduces all the blocks around the diagonal, which correspond to musical sections, to a zero value. We discuss the form to choose when applying an opening filter to extract information from the self-distance matrix: a constant square-shaped filter. But the size can also be adjusted to detect different musical structures. A large opening identify the global part of the piece while a smaller one reveal shorter sections. This idea is illustrated by detecting different musical structures in Edvard Grieg's *March of the Dwarfs* and Mozart's Piano Sonata *Alla Turca*.

4.2 Grayscale Mathematical Morphlogy

In this section, we define the principal operators of grayscale mathematical morphology that are useful in this chapter. This section generalizes the notions of binary mathematical morphology covered in Section 1.2.

4.2.1 Dilation and Erosion

In this chapter, we rely on mathematical morphology defined on functions, typically used to analyze gray level images, making an analogy between self-distance matrices and images. Only the useful notions are recalled here, and more details can be found in [Serra 1982, Heijmans 1990, Ronse 1991, Heijmans 1994, Bloch 2007, Najman 2010]. Let $(\mathcal{F}, \leq)$ be a lattice of functions (we consider here functions from $E = \mathbb{Z}^n$ into $\mathbb{R}^+$ to handle self-distance matrices, and the lattice is complete). Therefore, let $F \in \mathcal{F}$ (i.e. $F : E \to \mathbb{R}^+$), which can be considered as a grayscale image, and $S \in \mathcal{F}$ the structuring element, also called the structuring function, which respectively replace the sets X and S of Section 1.2. These functions have a $-\infty$ value outside their bounded support. To illustrate principal operations of grayscale mathematical morphology, we represent two functions F and S in Figure 4.1. In this example, the structuring function S is flat with the origin defined in the middle, which is a common choice for structuring functions.



(a) Grey-level function F (b) Structuring function S

Figure 4.1: Example of a function F (left) and a flat structuring element S (right).

Dilation $\oplus$ and *erosion* $\ominus$ in the complete lattice $(\mathcal{F}, \leq)$ are extensions of *Minkowski addition* [Minkowski 1903] and *subtraction* [Hadwiger 1950] in the binary morphological case, and are defined in Definition 4.1.

Definition 4.1: Dilation and Erosion

Let $F, S \in \mathcal{F}$, the **dilation** $\delta_S(F) = F \oplus S$ and **erosion** $\varepsilon_S(F) = F \ominus S$ of F by S are defined by:

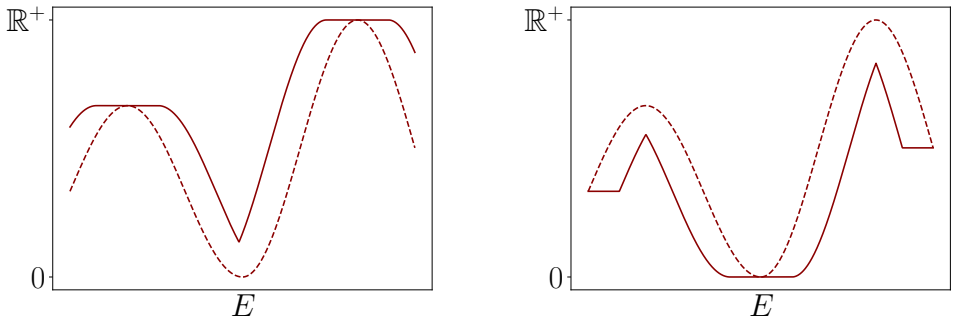
$$\forall x \in E, \quad F \oplus S(x) = \sup_{t \in E} \left(F(t) + S(x - t)\right)$$

$$\forall x \in E, \quad F \ominus S(x) = \inf_{t \in E} \left(F(t) - S(t - x)\right)$$

With the conventions that $\forall x \in \mathbb{R}, x + \infty = \infty$ and $x - \infty = -\infty$, which allows us to give a value to $F(t) + S(x - t)$ and $F(t) - S(t - x)$ when outside the domains of definition of the functions. Figure 4.2 illustrates these two operations by using the examples of the two functions defined in Figure 4.1. Because the origin is in the middle of the structuring function S, the following inequalities are satisfied:

$$\forall x \in E, \quad \varepsilon_S(F)(x) \leq F(x) \leq \delta_S(F)(x)$$

Applied to grayscale images, dilation extends bright zones and reduces dark ones, while erosion does the opposite (considering that bright pixels correspond to high values). As with binary morphology, these operations are non-reversible and non-linear. We do not mention any more properties, as this is not useful in the rest of this chapter, but these operations also satisfy the properties defined for the binary case in Section 1.2.



(a) Dilation (b) Erosion

Figure 4.2: Dilation and erosion of F by S (as defined in Figure 4.1) represented by the solid lines. To facilitate the understanding of these transformations, the F function is also represented by a dotted line.

4.2.2 Opening and Closing

As with binary mathematical morphology, the other two principal operations result from the composition of these operators. The *opening* $\circ$ is the composition of an erosion and a dilation and the *closing* $\bullet$ is a dilation followed by an erosion.

Definition 4.2: Opening and Closing

Let $S \in \mathcal{F}$, the **opening** γ_S and the **closing** φ_S by S are defined by:

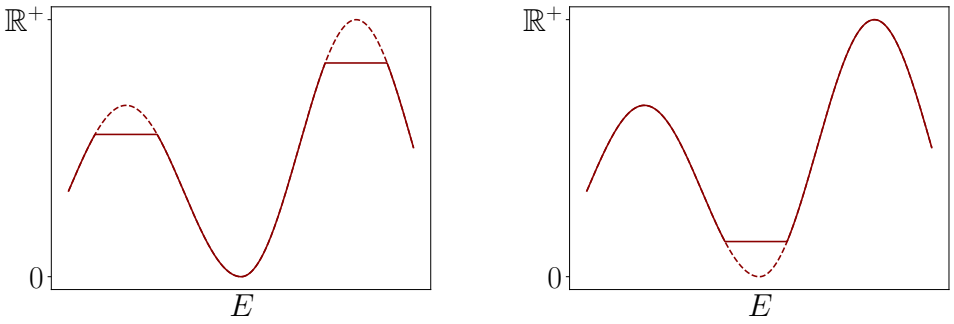
$$\begin{array}{lccl} \gamma_S : & \mathcal{F} & \longrightarrow & \mathcal{F} \\ & F & \longmapsto & F \circ S = (F \ominus S) \oplus S \\ \\ \varphi_S : & \mathcal{F} & \longrightarrow & \mathcal{F} \\ & F & \longmapsto & F \bullet S = (F \oplus S) \ominus S \end{array}$$

The result of these two operations is illustrated in Figure 4.3 with the F and S functions defined in Figure 4.1. Similarly to binary morphology, the following inequalities are always true:

$$\forall x \in E, \quad \gamma_G(F)(x) \leq F(x) \leq \phi_G(F)(x)$$

Opening and closing are increasing and idempotent operators, hence morphological filters. They can be used to eliminate small details (having higher values than their surrounding using opening, and smaller ones using closing) according to the size and shape of the structuring element. Therefore, by using these filters, some detailed information may be lost, while more flat and homogeneous regions

are obtained. This property will be used to highlight homogeneous regions in the self-distance matrix, in order to exhibit the main musical structures, as detailed in the remainder of this chapter.



(a) Opening (b) Closing

Figure 4.3: Opening and closing of F by S (as defined in Figure 4.1) represented by the solid lines. To facilitate the understanding of these transformations, the F function is also represented by a dotted line.

4.3 Self-Distance Matrix From Symbolic Music

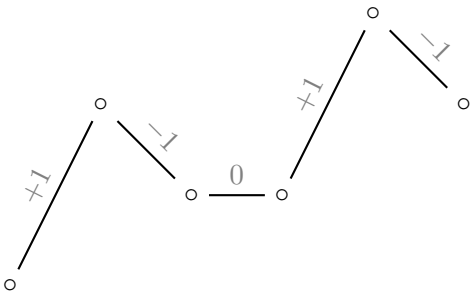
In this section, we define self-distance matrices and self-similarity matrices. We propose to generalize the melodic contour with the chord contour, and define a distance between chord contours to obtain self-distance matrices from symbolic representations of music.

4.3.1 Converting Symbolic Music to Sequence Using Chord Contour

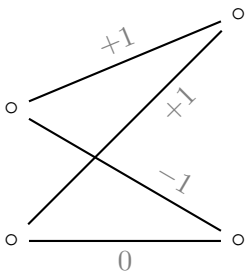
Whether for music perception [Trehub 1984, Dowling 1994], music analysis [Adams 1976] or music theory [Buteau 2008], *melodic contour* has become a fundamental tool in the music information research community. This tool applies on monophonic structures, i.e., musical phrases or motives in which two notes never sound at once. It is defined by the set of the directions between consecutive pitches of a melody, +1 and -1 indicating respectively an ascending and a descending interval. Figure 4.4(a) illustrates this idea by representing each note of a melody by a circle in a time/pitch graph. Melodic contour summarizes intervallic information and can be used to compare and classify melodic patterns or to help understand their perception. Considering the importance of melodic contour, it is not surprising that multiple extensions have been proposed. For example, two other contours were defined in [Anagnostopoulou 2013]: the *strong contour* (melodic contour of only the notes present on the beat) and the *weak contour* (strong contour with extra information if there is a contour variation within the beat). Moreover, it was proposed

in [Marvin 1987, Quinn 1999] to observe the directions at longer range, i.e., all the directions between the i^{th} and j^{th} pitches, not only between the i^{th} and $(i+1)^{th}$ pitches as for the usual melodic contour. To this purpose, both works used a matrix representation: *Morris's comparison matrix (COM-matrix)* in [Marvin 1987], and *combinatorial contour matrix* in [Quinn 1999]. In the COM-matrix, the coefficient at position (i, j) is the pitch direction between notes i and j, and for the combinatorial contour matrix this coefficient is $+1$ if the j^{th} note is higher in pitch than the i^{th} note or 0 otherwise. However, these generalizations remain in the monophonic context, and they do not handle musical chords.

We propose a generalization of the melodic contour to chord sequences, i.e., not restricted to note sequences. In the proposed definition, the direction between the pitches of two given chords is no longer a number but a matrix, called *chord contour*. The coefficient (i, j) of the chord contour is the direction between the i^{th} note of the first chord and the j^{th} note of the second chord, where the notes of the chords are ordered in descending pitch order. Therefore, the chord contour from an n-note chord to an m-note chord is of size $n \times m$. Figure 4.4(b) illustrates the construction of the chord contour: in this example, the two chords have two notes, so the corresponding chord contour is a 2×2 matrix. The first row corresponds to the directions from the highest note of the first chord to the notes of the next chord, and so on.



(a) The melodic contour is $\{1, -1, 0, 1, -1\}$



(b) The chord contour is $\begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}$

Figure 4.4: Illustration of the melodic contour and the chord contour.

The chord contour sequence of the introduction of Edvard Grieg's *March of the Dwarfs* is graphically represented in Figure 4.5. It will be analyzed in the following in order to identify the main passages or blocks of this sequence.

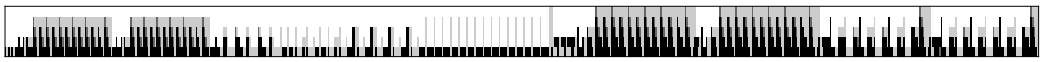


Figure 4.5: Representation of the chord contour sequence of the introduction of *March of the Dwarfs*. Black, dark gray and light gray pixels map respectively to values of 1, 0 and -1.

4.3.2 Distance Between Chord Contours

We propose here to define a distance between two chord contours. The main difficulty comes from the fact that chord contours, which are matrices, can have different sizes. First, we consider two chord contours with the same size. In this case, the *Hamming distance* is used. Let $A = (a_{i,j})$ and $B = (b_{i,j})$ be two chord contours of sizes $n \times m$, the Hamming distance $d(A, B)$ between the matrices A and B is defined as the number of coefficients which differ:

$$d(A, B) = |\{(i, j) \in [1...n] \times [1...m] \mid a_{i,j} \neq b_{i,j}\}|$$

If one of the two matrices has more rows (or columns) than the other matrix, one can reduce it by deleting rows (or columns) in order to get two matrices of the same size and use the previous formula, with the addition of the number of deleted rows (or columns). The rows (or columns) to be deleted are those that minimize the distance between the two matrices. Deleting a row (respectively a column) corresponds to omitting a note in the first chord (respectively the second chord). Thus, if A and B are two matrices of size $n_1 \times m_1$ and $n_2 \times m_2$, the distance $\mathcal{D}(A, B)$ between these two matrices is defined as:

$$\mathcal{D}(A, B) = \min_{A', B'}(d(A', B')) + |n_1 - n_2| + |m_1 - m_2|,$$

where A' and B' are two matrices of size $\min(n_1, n_2) \times \min(m_1, m_2)$ such that A' (respectively B') is obtained by removing $n_1 - \min(n_1, n_2)$ rows and $m_1 - \min(m_1, m_2)$ columns from A (respectively B). From a mathematical point of view, the first distance d respects symmetry, identity of indiscernibles, non-negativity and triangular inequality. It is well defined as a metric on the space of matrices with the same size. On the other hand, for the second distance $\mathcal{D}$, the triangular inequality is lost; hence, it is only a semi-metric in the mathematical sense. However, since we only make pairwise comparisons, without looking for a path from one matrix to another one in the space of matrices, the triangular inequality is not essential.

4.3.3 Self-Distance Matrix

In order to visualize the musical structures, the *self-similarity matrix* was proposed by Foote [Foote 1999], as a two-dimensional representation defined by computing the similarity between any two instants. Self-similarity matrices have become a fundamental concept in the study of musical structures [Paulus 2010]. In addition, the dual of self-similarity matrices are *self-distance matrices*, similarity can be obtained from a distance, and vice versa. Given an ordered sequence, the coefficient of the line i and the column j of the self-distance matrix (respectively the self-similarity matrix) is defined by the distance (respectively the similarity) between elements i and j of this sequence. We focus here on self-distances matrices, but the same logic can be transcribed to self-similarity matrices. Let c_k be the k^{th} chord contour of the musical piece, i.e., from the k^{th} chord to the $(k + 1)^{th}$ chord. Therefore, the

coefficient of the line i and the column j of the self-distance matrix is defined by $\mathcal{D}(c_i, c_j)$. Figure 4.6 displays the self-distance matrix corresponding to the example of the introduction of *March of the Dwarfs* in Figure 4.5. Since $\mathcal{D}$ is symmetric, the self-distance matrix is a symmetric matrix. The musical structures can be inferred from the information near the diagonal: the different blocks around the diagonal framed in red in Figure 4.6 represent the musical sections. It is possible to understand the shape of the self-distance matrix in comparison to the chord contour sequence: blocks on the diagonal correspond to sections that are visually identifiable in Figure 4.5.

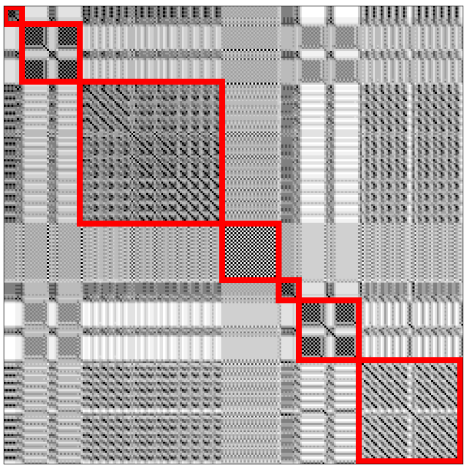


Figure 4.6: Self-distance matrix of the introduction of *March of the Dwarfs* (black = 0, i.e. low distance and high similarity, white = high distance values and low similarity).

In this section, we have proposed a method for converting symbolic music data into a self-distance matrix. Other solutions are possible without using the chord contour. However, this method is sufficient to illustrate the important results of this chapter on the advantages of morphological filters for homogenizing the regions in the self-distance matrix.

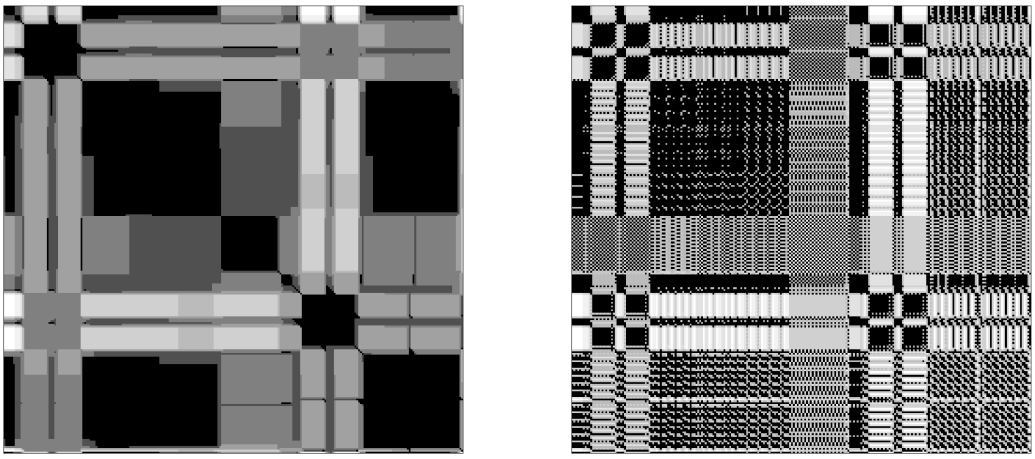
4.4 Application of Mathematical Morphology to the Self-Distance Matrix

In this section, we demonstrate the relevance of filtering the self-distance matrix using a morphological opening with a constant square-shaped structuring element to detect blocks around the diagonal. We also point out that these results can be obtained on a self-similarity matrix, by applying morphological closing. Finally, we demonstrate that changing the size of the morphological filter produces different structures in a hierarchical way.

4.4.1 Application of an Opening Filter to a Self-Distance Matrix

We propose here to use morphological filters to identify the main blocks of the self-distance matrix. Blocks along the diagonal provide information on the musical structure of the piece since low distance values of the self-distance matrix correspond to passages with high similarity. In order to identify similar blocks, locally higher distance values should be removed. The opening operation is particularly well suited to this situation. To do this, the structuring element has to be constant and square-shaped in order to preserve the general organization of the matrix, which exhibits strong vertical and horizontal structures, as well as squared blocks. By using this operation, it is possible to homogenize the regions of the self-distance matrix and to reduce the blocks on the diagonal to a zero value (because the diagonal coefficients are equal to zero due to the identity of indiscernibles of the metric).

The result of the opening operation on the self-distance matrix of Figure 4.6 with a square structuring element of size 12×12 is represented in Figure 4.7(a). Blocks on the diagonal appear in black, which is the minimal value (equal to zero), and we can easily detect them. To compare this method with simpler methods, thresholding is shown in Figure 4.7(b). Here, each coefficient below half of the maximum coefficient of the matrix is set to zero. Therefore, the thresholding does not detect the main blocks of the self-distance matrix. The threshold operation acts globally on the matrix, with the same threshold value applied everywhere. By contrast, opening is an operator that acts locally on the coefficients of the matrix, depending on local shape and size of the distance function, not on absolute values, which fits our filtering objective better.



(a) Result of an opening filter

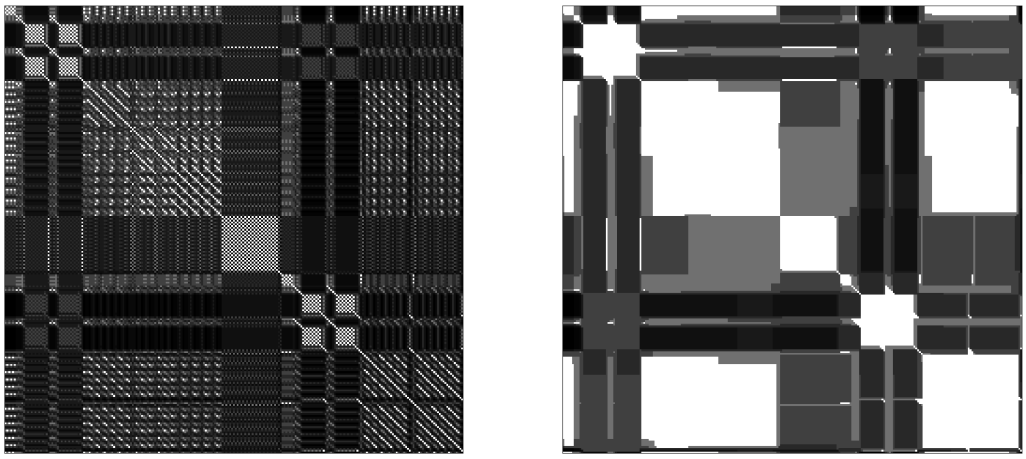
(b) Result of a thresholding

Figure 4.7: Filtering of the self-distance matrix using a morphological opening (a). As a comparison, a simple thresholding is shown in (b). The initial self-distance matrix is displayed in Figure 4.6.

4.4.2 Application of a Closing Filter to a Self-Similarity Matrix

The *self-similarity matrix* [Foote 1999], is also used in audio-based approaches to the analysis of musical structures. In this case, the values of the self-similarity matrix are inverted with respect to the self-distance matrix. The diagonal coefficients are the highest values (equal to one) and the goal is to remove locally lower values and to reduce the blocks around the diagonal to the highest value of the matrix. This change can also be handled with the morphological tools because dilation and erosion (respectively opening and closing) form pairs of dual operators. This means concretely that applying an opening on a self-distance matrix is equivalent to applying a closing on a self-similarity matrix, and vice versa.

For example, we generated a self-similarity matrix from the self-distance matrix shown in Figure 4.6 by defining the coefficient (i, j) of the self-similarity matrix as $\frac{1}{1+\mathcal{D}(c_i,c_j)}$. Other transformations can be used between distance and similarity, but this one is sufficient to illustrate the concept of duality between filtering a self-distance matrix by a morphological opening and filtering a self-similarity matrix by a morphological closing. With this definition, the self-similarity matrix is illustrated in Figure 4.8(a) with the same color code as the previous figures. Using a filtering by a morphological closing, we can then remove the low local values to isolate the blocks around the diagonal as illustrated in Figure 4.8(b) with a closing by a structuring element of size 12×12. In this case, these blocks are reduced to the local maximum value, in this case the one value, as this is the maximum similarity value on the diagonal. Therefore, similar results presented in this chapter can also be obtained on self-similarity matrices with closing filters.



(a) Self-similarity matrix (b) Result of an closing filter

Figure 4.8: Self-similarity matrix (a) and filtering by morphological closing (b). This demonstrates the duality between filtering a self-distance matrix by a morphological opening and filtering a self-similarity matrix by a morphological closing.

4.4.3 Changing the Shape of the Morphological Filter to Detect Different Musical Structures

The morphological operations provide new computational tools for the analysis and identification of the overall structure of a musical piece. Moreover, it is possible to detect musical structures at different scales, for example to refine the granularity of the analysis and identify the bars of the piece. This can be done by changing the size of the structuring element, in order to detect blocks of different sizes. With a smaller structuring element, it is possible to detect smaller blocks around the diagonal, representing for instance the bars of the piece, while a larger one discovers the global musical structure at a bigger scale.

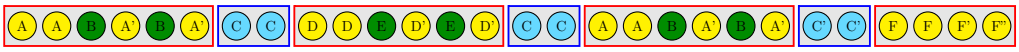
To illustrate the notion of filtering with different structuring elements, we consider the third movement of the Piano Sonata No.11 in A Major, composed by Wolfgang Amadeus Mozart and commonly known as *Alla Turca* or *Turkish Rondo*. The structures of the piece are represented in Figure 4.9(a), where each letter symbolizes 8 bars. This piece is divided into four main parts represented by red rectangles and linked with blue rectangles. There are two levels of structure: the 7 colored rectangles (global structure) or the 28 letters (detailed structure). As seen previously, the structuring element has to be constant and square-shaped, the only parameter to choose being the size. We applied an opening filter with a constant square-shaped structuring element of size 3×3 and 6×6 to the self-distance matrix (computed using the chord contour sequence). The result of these opening filters is displayed in Figures 4.9(b) and 4.9(c). For a clearer understanding, only the diagonal blocks (detected with the flood-fill algorithm) are shown in black in this figure, i.e., zero value coefficients connected to the diagonal of the matrix. We computed the *novelty score*, introduced by Foote [Foote 2000], of these two opening diagonals. The novelty score N is the correlation along the diagonal of a matrix M with the checkerboard kernel C:

$$N(t) = \sum_{i=-L/2}^{L/2} \sum_{j=-L/2}^{L/2} C(i,j)M(i+t, j+t),$$

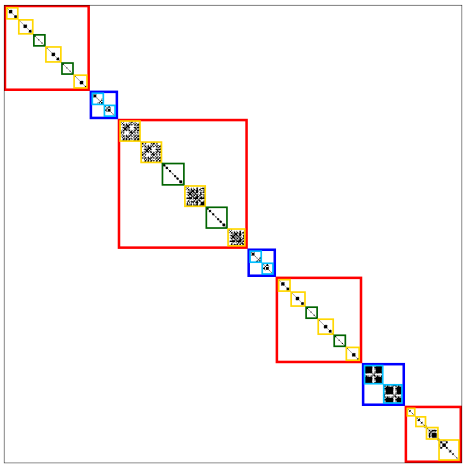
where C is the 64×64 symmetric matrix defined as:

$$C = \begin{pmatrix} -1 & \cdots & -1 & 1 & \cdots & 1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -1 & \cdots & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & -1 & \cdots & -1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & -1 & \cdots & -1 \end{pmatrix}$$

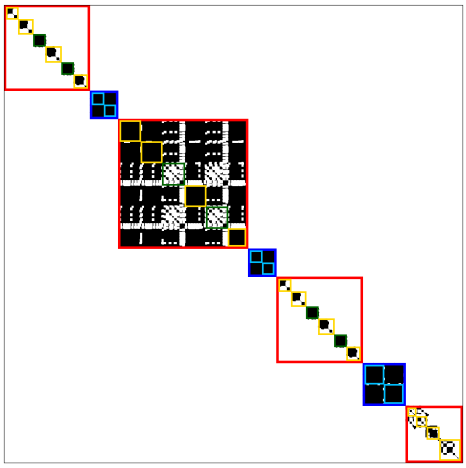
Notice that we use the opposite of the original checkerboard kernel [Foote 2000], because we have a self-distance matrix instead of a self-similarity matrix. This novelty score allows detecting changes, and therefore the limits of the blocks to identify. The novelty scores of the two opening diagonals are represented in Figure 4.9(d) and



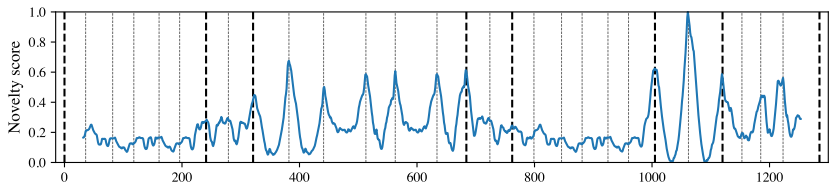
(a) Musical structures of *Alla Turca* (W.A. Mozart)



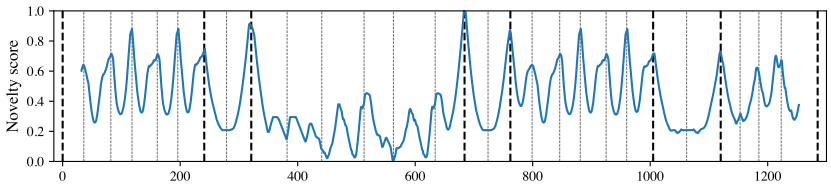
(b) Opening diagonal with a 3×3 constant square-shaped structuring element



(c) Opening diagonal with a 6×6 constant square-shaped structuring element



(d) Novelty score of the opening diagonal with a constant square-shaped 3×3 structuring element



(e) Novelty score of the opening diagonal with a constant square-shaped 6×6 structuring element

Figure 4.9: Filtering of the self-distance matrix at different scales by the morphological opening in order to obtain different musical structures in a hierarchical way.

Figure 4.9(e). We also add the boundaries of the musical structures shown in Figure 4.9(a) with thick dotted lines (boundaries between rectangles) and thin dotted lines (boundaries between letters). The high value of the novelty score represents the boundaries of the piece. The novelty score of the opening diagonal with a 3×3 structuring element detects the boundaries between the D/D/E/D'/E/D' and C'/C'

sections. While the novelty score of the opening diagonal with a 6×6 structuring element detects the boundaries between the rectangles and the A/A/B/A'/B/A' sections. With these two diagonals blocks, it is possible to detect two different structures of the piece.

Finally, we can adjust the size of the structuring element used to filter the self-distance matrix depending on the granularity that we want in the analysis of the musical structures (which enables for example to detect only few very long passages or a greater number of short passages). By varying the size of the structuring element, we can computationally grasp the segmentation process at multiple levels. In fact, every time we increase the size of the structuring element we force some segments to merge and become a new bigger segment, starting from few notes segments to the whole piece.

4.5 Conclusion

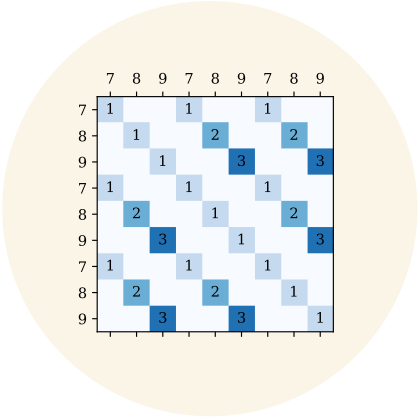
Mathematical morphology is a relatively uncommon theory in the music information research community. However, we have already revealed in Part I that binary mathematical morphology is relevant to symbolic representations of music whether for analysis or generation. In addition, we have demonstrated in this chapter that the grayscale framework of mathematical morphology can be successfully applied to discover musical structures by filtering the self-distance matrix. In particular, these morphological filters have been used to homogenize and identify well-defined regions of the self-distance matrix corresponding to musical sections. The opening operation has been applied to the analysis of the musical structures of a piece because it locally removes the high values of the self-distance matrix. With a constant square-shaped structuring element, it reveals the horizontal and vertical blocks of the self-distance matrix. In addition, the blocks around the diagonal, which correspond to a well-defined musical structure, all have a zero value. Moreover, by varying the size of the filter, it is possible to have different filtering levels in the automatic detection of the underlying structures of the musical piece. By filtering the self-distance matrix with a large opening, one is able to identify the main global parts of the piece, while using a smaller morphological filter reveals shorter musical sections. We have also demonstrated that similar results can be obtained with self-similarity matrices using morphological closing filters. Some promising results of applying this new method in the field of music automatic segmentation have been obtained and discussed by presenting a computational analysis of an excerpt of Edvard Grieg's *March of Dwarfs* and of Mozart's Piano Sonata *Alla Turca*.

In this chapter, we demonstrated the usefulness of morphological filters to homogenize musical sections to detect the musical structure. However, homogeneity is not the only criteria for music structure analysis, and the other main criterion is repetition. Paulus et al. argue that a combined approach (based on homogeneity, novelty and repetition) provides promising results [Paulus 2010]. Our method does not handle repetition, because the goal of this chapter is to show the application of

mathematical morphology for music structures analysis. Due to the the simplicity yet powerful utility of morphological filters, we strongly believe that this method can be reuse for future algorithms for the homogeneity step. Moreover, although we have applied this method on symbolic music representations with a chord contour sequence, this method can also be applied for audio-based analysis of musical structures. For future research, we believe that it is relevant to test this method on a large audio database with hierarchically annotated structures in order to validate it experimentally. It may also be pertinent to combine this approach with the repetition criterion, as stated by Paulus et al. [Paulus 2010], for example by using the method proposed in the next chapter, which is based on repetition .

Chapter 5

Repetition-Based Algorithm for Generating Hierarchical Segmentations of Music



Contents

In this chapter, we present an algorithm that can generate multiple hierarchical segmentations of a musical sequence based on approximate repeated patterns. A similarity between music objects defines this approximation to obtain an Adapted Correlative Matrix, which reveals repeated patterns without overlaps. Based on this representation, we propose an algorithm that extracts meaningful information to identify segmentations in a hierarchical way. Finally, we demonstrate that changing the relation produces different hierarchical segmentations of the same sequence and provides a possible explanation of the obtained segmentations.

Section 5.2 generalizes the existing definition of the Correlative Matrix and introduces the Adapted Correlative Matrix in order to work with non-overlapping repeating patterns. Section 5.3 describes the proposed algorithm which extracts meaningful information of the Adapted Correlative Matrix to generate hierarchical segmentations. Section 5.4 illustrates this method with various music objects, starting with a sequence of notes, then a sequence of chords, and finally with a sequence of bars.

This chapter is an extended version of the article "Generating Multiple Hierarchical Segmentations of Music Sequences using Adapted Correlative Matrices" [Lascabettes 2022b].

5.1 Introduction

Segmentation is an important problem for music analysis, performance, perception, and retrieval. It consists in dividing a musical sequence into non-overlapping segments. Music segmentation has been studied in the audio and symbolic domains. However, compared to work on audio sources, there has been comparatively less work on segmentation in the symbolic domain [Giraud 2016]. Segmentation tasks with symbolic sources focus on the musical score. Lerdahl and Jackendoff proposed the *Generative Theory of Tonal Music* (GTTM) [Lerdahl 1985], where they started to model segmentations in a hierarchical way. This was expanded to include aspects of harmonic tension in *Tonal Pitch Space* [Lerdahl 2001]. In parallel, computational approaches based on the GTTM rules were developed [Temperley 2004]. Other algorithms, such as Chew's *Boundary Search Algorithm*, were based on tonality, using key boundaries to create segmentations [Chew 2002]. More recently, the *Correlative Matrix* was used to first discover and classify patterns, then determine the best segmentation using a score function [Rafael 2010]. However, in the definition of the Correlative Matrix, patterns may overlap, which is incoherent with the segmentation task. Therefore, we propose to define the *Adapted Correlative Matrix* to discover repeated patterns without overlaps in a musical sequence.

We develop in this chapter an algorithm that generates hierarchical segmentations of a musical sequence based on approximate repeated patterns. The chosen approximation is defined by a relation between music objects which generates the

Adapted Correlative Matrix. Our algorithm iteratively selects repeated patterns based on their distinctiveness, that is to say if other repeated patterns begin at the same time than the starting note (reinforcing the beginning boundary) or immediately after it (reinforcing the end boundary). The distinctiveness criterion exploits the Adapted Correlative Matrix structure by ensuring that the selected pattern is compatible with other repeated patterns in a hierarchical way. Assuming that the beginning and the end of repeated patterns influence the segmentation of a musical sequence [Cambouropoulos 2006], our algorithm iteratively creates boundaries to obtain hierarchical segmentations. We apply this method to different musical genres and music objects: a sequence of notes, chords and bars. In each case, we define several relations between music objects which yield different hierarchical segmentations. Finally, we visualize the results obtained with a tree representation and discuss the results.

5.2 Definition of the Adapted Correlative Matrix

In this section, we propose a definition of the Correlative Matrix that generalizes previous applications, and we then introduce the Adapted Correlative Matrix for discovering patterns in a musical sequence without overlaps.

5.2.1 Correlative Matrix

The *Correlative Matrix* was first introduced in music processing in order to discover exact repeating patterns in a sequence of pitches [Hsu 1998, Hsu 2001]. For a given sequence of pitches $(p_1, \ldots, p_n)$ of length n, the first definition of the Correlative Matrix was an $n \times n$ matrix, where the coefficient of the i^{th} row and the j^{th} column is set to 1 if $p_i = p_j$. Moreover, if $p_i = p_j$ and $p_{i+1} = p_{j+1}$, meaning there is one or more repeated patterns of length two, the coefficient of the $i + 1^{th}$ row and the $j + 1^{th}$ column is set to 2. By by iteratively applying this process, the value of each coefficient of the Correlative Matrix indicates the length of a repeating pattern. Compared to the Self-Similarity Matrix used in audio-based music segmentation [Foote 1999], the Correlative Matrix allows us to easily discover the longest repeating patterns with the maximal coefficients. Later, the Correlative Matrix has also been defined to allow for a sequence of intervals or a combination of pitch contours and note durations [Rafael 2010]. Therefore, the coefficient of the Correlative Matrix was modified to check if two elements of the sequence were equal (with regard to pitch) [Hsu 1998, Hsu 2001] or below a similarity threshold (with regard to pitch, contour and duration) [Rafael 2010]. This can be generalized with a symmetric and reflexive relation[1] in order to capture a wide variety of musical objects, indeed the equality or the similarity threshold are both symmetric and reflexive relations. We then propose to generalize the Correlative Matrix with the following definition:

[1] Let $T = (t_1, \ldots, t_n)$ be a sequence. A relation $\equiv$ on T is symmetric if: $\forall t_i \in T,\ t_i \equiv t_i$ and reflexive if: $\forall t_i, t_j \in T,\ t_i \equiv t_j \Leftrightarrow t_j \equiv t_i$.

Definition 5.1: Correlative Matrix (CM)

Let $T = (t_1, \ldots, t_n)$ be a sequence and $\equiv$ a symmetric and reflexive relation on T. The **Correlative Matrix** (CM) is an $n \times n$ matrix where the coefficient $C_{i,j}$ of line i and column j is defined by:

$$C_{i,j} = \begin{cases} C_{i-1,j-1} + 1, & \text{if } t_i \equiv t_j, \\ 0, & \text{otherwise,} \end{cases}$$

with the convention: $C_{i,j} = 0$ if i or j is negative.

For example, if $T = (7,8,9,7,8,9,7,8,9)$ and $\equiv$ is the usual equality $=$ on $\mathbb{R}$ (i.e. $7 \equiv 7$ and $7 \not\equiv 8$), the Correlative Matrix is represented in Figure 5.1. The following patterns are discovered: $(7,8,9,7,8,9,7,8,9)$, $(7,8,9,7,8,9)$ and $(7,8,9)$. The first two patterns are discovered in the sequence because overlaps are allowed. In order to use the idea of the Correlative Matrix to identify contiguous segmentations of T, we need to adapt the definition of the Correlative Matrix to disallow overlaps. Granted, in music, there exist cases where the last note of a segment can be the first note of the next [Lerdahl 1985], but this is very rare and therefore outside the scope of this work.

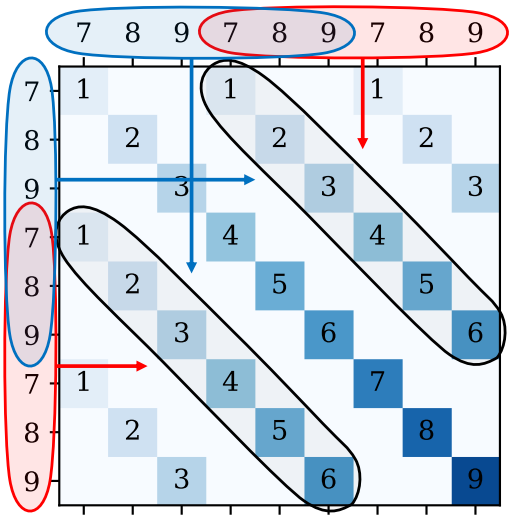


Figure 5.1: Correlative Matrix generated by the sequence $T = (7,8,9,7,8,9,7,8,9)$.

5.2.2 Adapted Correlative Matrix

Here, we define the *Adapted Correlative Matrix* where approximate repeated patterns are discovered in a sequence T without overlaps.

Definition 5.2: Adapted Correlative Matrix (ACM)

Let $T = (t_1, \ldots, t_n)$ be a sequence and $\equiv$ a symmetric and reflexive relation on T. The **Adapted Correlative Matrix** (ACM) is an $n \times n$ matrix where the coefficient $C_{i,j}$ of line i and column j is defined by:

$$C_{i,j} = \begin{cases} C_{i-1,j-1} + 1, & \text{if } t_i \equiv t_j \text{ and } C_{i-1,j-1} + 1 \leq |i-j|, \\ 1, & \text{if } t_i \equiv t_j \text{ and } C_{i-1,j-1} + 1 > |i-j|, \\ 0, & \text{otherwise,} \end{cases}$$

with the convention: $C_{i,j} = 0$ if i or j is negative.

When $t_i \equiv t_j$, the maximal length of the pattern without overlaps is $|i-j|$, as with any longer pattern, the higher index would be in both occurrences of the pattern. For example, if $t_3 \equiv t_6$, the maximum pattern length without overlaps is $|3-6| = 3$. Therefore, if $C_{i,j} = |i-j|$ and $t_{i+1} \equiv t_{j+1}$, instead of continuing to increment the value for $C_{i+1,j+1}$ and discovering overlapping patterns, in the ACM we restart at $C_{i+1,j+1} = 1$. For instance, the ACM for the sequence $T = (7,8,9,7,8,9,7,8,9)$ and $\equiv$, the equality on T, is presented in Figure 5.2.

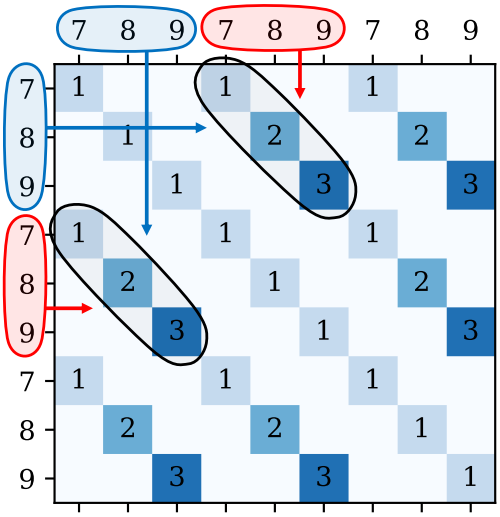


Figure 5.2: Adapted Correlative Matrix for the sequence $T = (7,8,9,7,8,9,7,8,9)$.

In Figure 5.2, the longest discovered pattern is now: $(7, 8, 9)$. Unlike the case with the Correlative Matrix, the pattern $(7, 8, 9, 7, 8, 9)$ is not discovered because this pattern is repeated with overlapping with $(7, 8, 9)$. The Correlative Matrix was successfully used to discover approximate repeated patterns in a musical sequence, allowing for overlaps. However, by defining the Adapted Correlative Matrix we can retain the fundamental idea of the Correlative Matrix while avoiding overlaps, which is required for the music segmentation task.

5.3 Algorithm for Extracting Hierarchical Segmentations Using the Adapted Correlative Matrix

In this section, we propose an algorithm to extract information from the Adapted Correlative Matrix in order to obtain hierarchical segmentations of a musical sequence.

5.3.1 Algorithm for Iteratively Discovering the Most Distinct Patterns

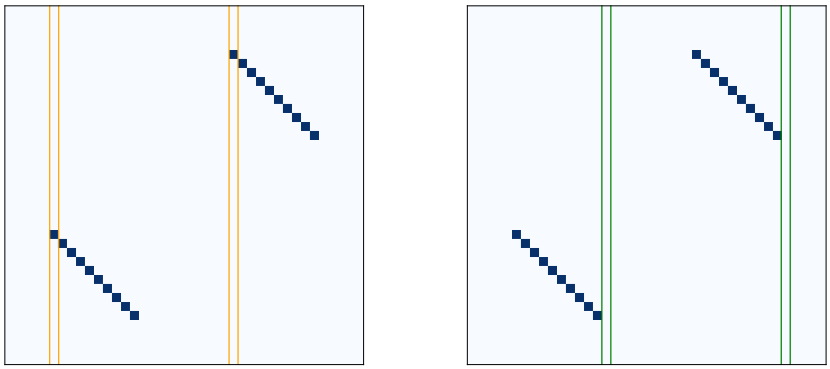
Let ACM be the Adapted Correlative Matrix of a sequence $T = (t_1, \ldots, t_n)$ with $\equiv$ a symmetric and reflexive relation. We describe here an algorithm that extracts data from the ACM in order to identify segmentations of T in a hierarchical way. This algorithm has three steps: it first selects the longest approximate repeating patterns, then the most distinct of these, before removing it from the data and repeating the process. The resulting hierarchy of segmentations is a natural consequence of this algorithm.

- **Step 1: Select the longest approximate repeating patterns**
  The *longest approximate repeating patterns* are defined by all the pairs $((t_{i-C_{i,j}}, \ldots, t_i), (t_{j-C_{i,j}}, \ldots, t_j))$, where $C_{i,j} = \max_{1 \leq k,l \leq n} C_{k,l}$ (the value of the maximal coefficients of the ACM). Often there will be more than one pattern tied for longest.

- **Step 2: Select the most distinct pair**
  Among all the discovered pairs from step 1, the *most distinct one* is the one that maximizes the number of repeating patterns that begin at the same time than the starting note (reinforcing the beginning boundary) or immediately after the two patterns of the pair (reinforcing the ending boundary). That is to say, we choose the pair that maximizes the number of coefficients equal to 1 in columns $i - C_{i,j}$ and $j - C_{i,j}$ as illustrated in Figure 5.3(a) (patterns that start at the same times as the pair) and in columns $i + 1$ and $j + 1$ illustrated in Figure 5.3(b) (patterns that start just after the pair).

- **Step 3: Remove the most distinct pair from the ACM**
  We then update the ACM by removing the most distinct pair and add bound-

aries to the segmentation at the beginning and end of the most distinct patterns. All coefficients of the most distinct pair become equal to 0 except for the first coefficient which remains equal to 1 (this is useful in step 2 for future iterations). Moreover, in order to have hierarchical segmentations, if a coefficient $C_{i',j'} > 1$ is on the same column or line as the beginning or the end of a pattern of the most distinct pair, this coefficient is equal to 1 and $C_{i'+k,j'+k}$ is 0, which is described in more detail in the next section. This creates a boundary that remains for the next segmentations and we thus obtain hierarchical segmentations. Finally, we go back to step 1 until there is no coefficient greater than 1 in the ACM.



(a) Discovery of patterns that start at the same time as the pair.

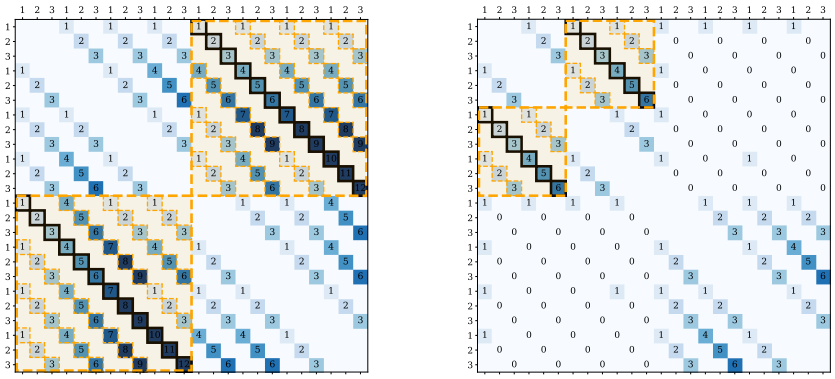
(b) Discovery of patterns that start just after the end of the pair.

Figure 5.3: Criteria for step 2 of the algorithm to choose the most distinct pair of patterns among the longest ones in the ACM.

5.3.2 Avoid Discovering Redundant Patterns

We describe here the condition in step 3 to avoid discovering redundant patterns that are not interesting for the segmentation task. To understand this condition, let us consider the sequence T composed of $1, 2, 3$ repeated eight times with $\equiv$ as the equality on T. The ACM of this sequence is shown in Figure 5.4(a) with $(1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3)$ (of length 12) as the longest repeating pattern because the first half of the sequence matches the second half. If the coefficients are not reset to 0, the second longest repeating pattern is $(1, 2, 3, 1, 2, 3, 1, 2, 3)$ (of length 9). However, this is rediscovering a pattern that comes from a match between the first and second half of the T sequence, and which is in fact a sub-pattern of the first discovered pattern. To avoid discovering such redundant sub-patterns, we add the condition in step 3 where some coefficitens are reset to 0 in the loop of the algorithm. In other words, when the most distinct pair $((t_{i-C_{i,j}}, \ldots, t_i), (t_{j-C_{i,j}}, \ldots, t_j))$ is discovered, we exclude the possibility of discovering sub-patterns resulting from a

match between the two patterns of the most distinct pair. To do this, we reset the coefficients $C_{p,q}$ and $C_{q,p}$ to 0 when $i - C_{i,j} \leq p \leq i$ and $j - C_{i,j} \leq q \leq j$. These correspond to the two square areas in the ACM that contain the most distinct pair, as represented in Figure 5.4 by the dashed orange areas. Coefficients greater than 1 are set to 0, while coefficients equal to 1 (meaning the starting point of a pattern) are unchanged as they are useful for step 2 of the algorithm (selecting the most distinct pair). This is illustrated in Figure 5.4(b) where some of the ACM coefficients have been reset to 0 while removing the first most distinct pair from the ACM.



(a) The most distinct pair has length 12 (the first half of the sequence matches the second half).

(b) Since some coefficients have been reset, the second most distinct pair is of length 6 (not 9).

Figure 5.4: Some coefficients are reset to avoid discovering sub-patterns that come from an already discovered pattern.

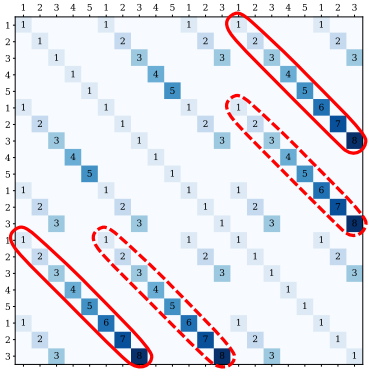
5.3.3 Application and Visualization of the Results

Let us take an example with the sequence of real numbers:

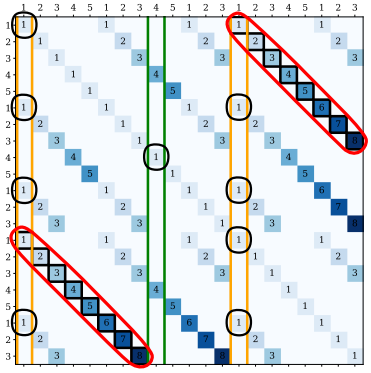
$$T = (1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 3, 1, 2, 3, 4, 5, 1, 2, 3),$$

with $\equiv$ as the equality on T. The resulting ACM is illustrated in Figure 5.5(a). The maximal coefficient is 8, and two pairs of longest repeating patterns are discovered and represented in Figure 5.5(b) and Figure 5.5(c). The pair that maximizes the number of patterns that start at the same time and right after the end of the pair is shown in Figure 5.5(c).

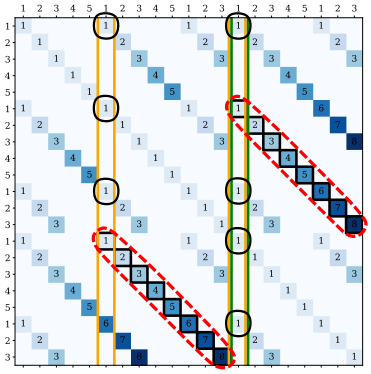
Hierarchical segmentations of musical notes have been studied by Lerdahl and Jackendoff based on different rules on the proximity between notes, repetitions and strong/weak accent of the rhythm in the GTTM [Lerdahl 1985]. They have represented hierarchical segmentations using a tree visualization. However, the method developed in this chapter handles music objects other than notes and is able to propose multiple hierarchical segmentations based on the chosen relation between these objects. Therefore, we also used a tree representation to visualize hierarchical



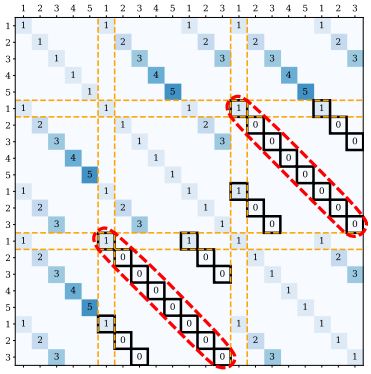
(a) **Step 1:** Two pairs of patterns of length 8 are discovered in the ACM.



(b) **Step 2:** For the first pair, 10 patterns start at the same time and 1 pattern starts after the end of the pair (total: 10+1=11).



(c) **Step 2:** For the second pair, 9 patterns start at the same time and 5 patterns start after the end of the pair (total: 9+5=14). This is the most distinct pair.



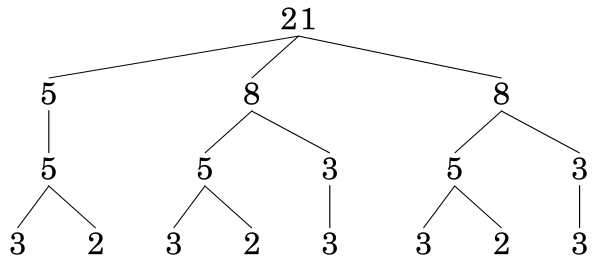
(d) **Step 3:** The most distinct pair is removed and boundaries are created at the dotted orange columns and lines (changes in black).

Figure 5.5: Illustration of the first loop of the algorithm. These three steps are iterated until the coefficients of the ACM are only 0 or 1.

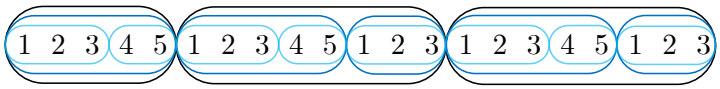
segmentations. By representing the length of the discovered patterns (the most distinct pairs), the tree visualization of the previous sequence T (with $\equiv$ the equality) is shown in Figure 5.6. In this case, our algorithm discovers four segmentations that are hierarchically structured. The first segmentation is the sequence itself, so the length of T (here 21) is represented at the top of the tree. The three other discovered segmentations of T are:

- (1,2,3,4,5),(1,2,3,4,5,1,2,3),(1,2,3,4,5,1,2,3) represented by the line 5/8/8;
- (1,2,3,4,5),(1,2,3,4,5),(1,2,3),(1,2,3,4,5),(1,2,3) represented by the line 5/5/3/5/3; and,

- (1,2,3),(4,5),(1,2,3),(4,5),(1,2,3),(1,2,3),(4,5),(1,2,3) represented by the line 3/2/3/2/3/3/2/3.



(a) Tree visualization of the hierarchical segmentations where the numbers indicate the length of each discovered repeating pattern.



(b) Hierarchical segmentations represented by grouping elements of the sequence T.

Figure 5.6: Visualization of the different segmentations in a hierarchical way of the sequence $T = (1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 2, 3, 1, 2, 3, 4, 5, 1, 2, 3)$.

5.4 Applications to Symbolic Music Representations

The algorithm presented in Section 5.3 generates hierarchical segmentations of a sequence with a relation which characterizes the chosen approximation for discovering when two patterns are nearly the same. In this section, we apply this algorithm to various symbolic music representations: a sequence of notes (Section 5.4.1), a sequence of chords (Section 5.4.2) and a sequence of bars (Section 5.4.3). In each case, we propose several relations which lead to multiple hierarchical segmentations and we illustrate the results using a tree visualization.

5.4.1 When T is a Sequence of Notes

Let T be a sequence of musical notes. There are many ways to define a musical note. For example, a note can be defined as a triplet (p, o, d), as in [Giraud 2012b], where p is the pitch, o the onset and d the duration of the note. This can also be enriched to five parameters (o, p, mp, d, v) [Meredith 2002a], where mp is the *morphetic pitch* [Meredith 1999] and v the voice where the note occurs. It is also possible to add other parameters such as the velocity, *general pitch interval representation* [Cambouropoulos 1996], etc. Among all the different parameters, we

first choose here to define a note by its *interval* $\Delta p_i = p_{i+1} - p_i$ (p_i being the pitch of the i^{th} note), then we have $T = (\Delta p_1, \ldots, \Delta p_{n-1})$. For example, let T be the sequence of intervals of the first 64 intervals of the Prelude in C major from the *Well-Tempered Clavier* by Johann Sebastian Bach (BWV 846). In this case, $T = (4, 3, 5, ..., 5, 4, -16)$. A key advantage of this representation is that it is invariant to transpositions. There are also different choices for the relation $\equiv$, because they are many ways to define similarity between intervals [Giraud 2015]. Here we use a similarity threshold [Rafael 2010] and an "up/down" relation which defines the melodic contour [Ghias 1995]. Let Δp_i and Δp_j be two intervals of T and $\lambda \in \mathbb{N}$, these two relations are defined as follows (note that they are both symmetric and reflexive):

- **Similarity Threshold Intervals Relation:**

$$\Delta p_i \equiv_{sim} \Delta p_j \Leftrightarrow |\Delta p_i - \Delta p_j| \leq \lambda$$

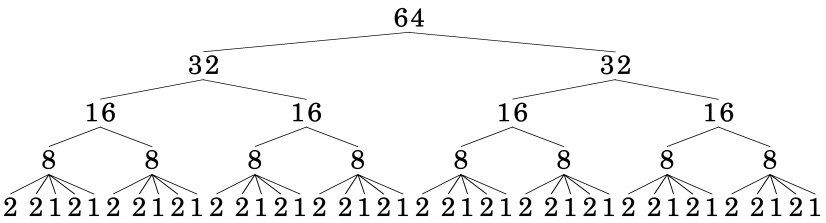
- **Melodic Contour Relation:**

$$\Delta p_i \equiv_{mc} \Delta p_j \Leftrightarrow \text{sgn}(\Delta p_i) = \text{sgn}(\Delta p_j),$$

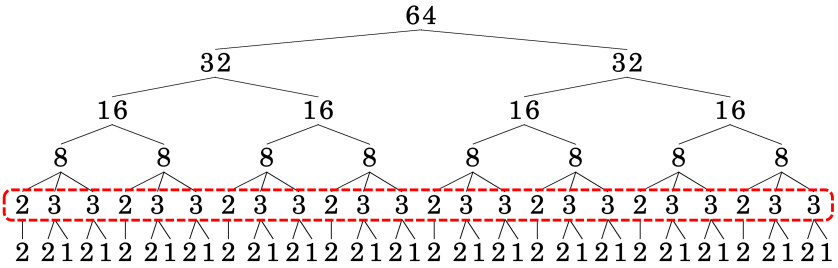
  where sgn is the sign function defined by $\text{sgn}(x) = 1, -1$ or 0 if $x > 0, x < 0$ or $x = 0$.

The results of our algorithm for these two relations are illustrated in Figure 5.7. We chose $\lambda = 4$ for the similarity threshold intervals relation to capture wide variations between intervals. The different hierarchical segmentations are represented as trees, and in the score where the different colors represent different segmentations.

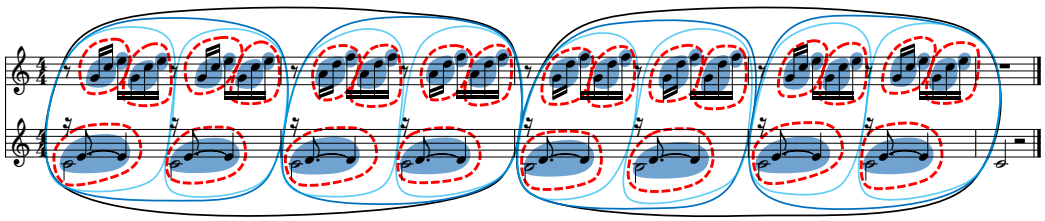
Since we chose $\lambda = 4$, the tolerance for two intervals to be equivalent with the similarity threshold intervals relation is set high, i.e. the patterns are discovered with a large approximation. As for the melodic contour, the two intervals simply need to have the same sign. This explains why the two hierarchical segmentations discovered in Figure 5.7(a) and Figure 5.7(b) are almost similar. However, the relation resulting from the melodic contour produces an additional segmentation, framed in dotted red in Figure 5.7(b) and on the score in Figure 5.7(c), which we can interpret. When looking at the first eight intervals of the T sequence, $(4, 3, 5, 4, -9, 5, 4, -16)$, the 5th and 8th intervals (-9 and -16 respectively) do not match with the similarity threshold intervals relation because $|(-9) - (-16)| = 7 > 4$. Since 4 is in relation to 5 and 3 is in relation to 4, patterns $(4, 3)$ and $(5, 4)$ are discovered. The similarity threshold intervals relation therefore yields the segmentation $(4, 3)(5, 4)(-9)(5, 4)(-16)$ indicated by the last line 2/2/1/2/1 in Figure 5.7(a). But, for the melodic contour relation, the two intervals -9 and -16 are in relation because they are both descending intervals. As a result, the patterns $(5, 4, -9)$ and $(5, 4, -16)$ are discovered, leading to the segmentation $(4, 3)(5, 4, -9)(5, 4, -16)$ which is indicated by the 2/3/3 line in Figure 5.7(b). This reasoning applies to the whole T sequence, not just the first eight intervals, and explains why the melodic contour reveals an additional



(a) Tree representation of the results based on the similarity threshold intervals relation for $\lambda = 4$.



(b) Tree representation of the results based on the melodic contour relation.



(c) Discovered hierarchical segmentations marked in the score for the two relations, where the blue color indicates common segmentations for both relations, while the dotted red color indicates the segmentation obtained only with the melodic contour relation.

Figure 5.7: Different trees of the same sequence of notes T (BWV 846) when the relation between intervals changes. The melodic contour relation provides an additional segmentation framed in dotted red.

segmentation. Moreover, these nearly repeated patterns discovered with the different relations could also be interesting for performance. For example, the identified patterns could be ones that might be highlighted through prosodic variations like dynamic accents; because the patterns are repeated and distinctive, they could also sound perceptually plausible.

5.4.2 When T is a Sequence of Chords

Let $T = (c_1, \ldots, c_n)$ be a sequence of chords where chords of root C are labelled from the set $\hat{\text{C}} = \{\text{C}, \text{C}^6, \text{C}^7, \text{C}^7_M, \text{C}_m, \text{C}^6_m, \text{C}^7_m, \text{C}^{M7}_m, \text{C+}, \text{C+}^7, \text{C}^{\text{o}}, \text{C}^{\text{o}7}, \text{C}^{\emptyset}\}$ and similarly for other roots. We could define the relation between two chords by strict

equality ($c_i \equiv_{st} c_j \Leftrightarrow c_i = c_j$) to have a strong constraint between chords, but this runs the risk of being overly sensitive to small ornamental changes.

Let us instead define the ChordType function of a chord $c_i \in \hat{\mathrm{C}}$ (and similarly for other roots) by:

$$\mathrm{ChordType}(c_i) = \begin{cases} \mathrm{C}_{maj} & \text{if } c_i = \mathrm{C}, \mathrm{C}^6, \mathrm{C}^7, \mathrm{C}^7_M, \\ \mathrm{C}_{min} & \text{if } c_i = \mathrm{C}_m, \mathrm{C}^6_m, \mathrm{C}^7_m, \mathrm{C}^{M7}_m, \\ \mathrm{C}_{aug} & \text{if } c_i = \mathrm{C}+, \mathrm{C}+^7, \text{ and} \\ \mathrm{C}_{dim} & \text{if } c_i = \mathrm{C}^{\mathrm{o}}, \mathrm{C}^{\mathrm{o}7}, \mathrm{C}^{\emptyset} \end{cases}$$

For example $\mathrm{ChordType}(\mathrm{C}^7_m) = \mathrm{C}_{min}$ or $\mathrm{ChordType}(\mathrm{F}\sharp^7) = \mathrm{F}\sharp_{maj}$. Let c_i and c_j be two chords of T, we can then define the ChordType relation $\equiv_{ct}$ on T by:

$$c_i \equiv_{ct} c_j \;\Leftrightarrow\; \mathrm{ChordType}(c_i) = \mathrm{ChordType}(c_j)$$

Some other relations, which we will not develop here, could include threshold relations based on the distance of chords within the Tonnetz [Krumhansl 1998] or one of the parsimonious relations defined by Douthett and Steinbach [Douthett 1998].

Let us take as example the song *In My Life* from The Beatles, released in 1965. All the songs from The Beatles are annotated at `http://isophonics.net` with: structural segmentation, key changes, chords, and beats. According to this database, the chords of the song are: T = (A, E, A, E, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, $\mathrm{F}_m\sharp$, D, G, A, $\mathrm{F}_m\sharp$, B, D_m, A, A, E, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, $\mathrm{F}_m\sharp$, D, G, A, $\mathrm{F}_m\sharp$, B, D_m, A, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, A, E, $\mathrm{F}_m\sharp$, A^7, D, D_m, A, $\mathrm{F}_m\sharp$, D, G, A, $\mathrm{F}_m\sharp$, B, D_m, A, A, E, D_m, A, E, A).

The results of the algorithm applied to this sequence T with the relation $\equiv_{ct}$ are represented in Figure 5.9. We can compare the hierarchical segmentations obtained by our algorithm with the annotated structure from the database represented in Figure 5.8.

```
0.000   0.416   :   silence
0.416   9.616   :   intro
9.616   28.302  :   verse
28.302  46.719  :   bridge
46.719  51.438  :   half-intro
51.438  70.206  :   verse
70.206  88.700  :   bridge
88.700  107.253 :   verse_(instrumental)
107.253 125.659 :   bridge
125.659 143.715 :   outro
143.715 147.973 :   silence
```

Figure 5.8: Annotated segmentation of *In My Life* from The Beatles where the three section of the piece are framed in dotted red.

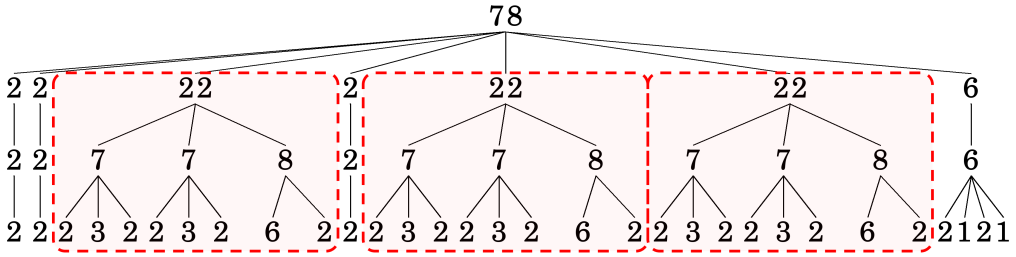


Figure 5.9: Hierarchical segmentations of The Beatles' *In My Life* based on the chord sequence where the three main sections framed in dotted red are correctly discovered.

We can observe that the second line (2/2/22/2/22/22/6) contains three main sections of length 22 framed in dotted red. Each of these sections corresponds to `Verse/Bridge` and the remaining sections of length 2 and 6 map to the `Intro`, `Half-intro` and `Outro`. In the third line, i.e. (2/2/7/7/8/2/7/7/8/7/7/8/6), the `Verse` section is subdivided into two sequences of 7 chords and the `Bridge` is a sequence of 8 chords. In this example, the algorithm correctly identifies the different segmentations because repetition is enough to discover the main sections.

5.4.3 When T is a Sequence of Bars

Let $T = (b_1, \ldots, b_n)$ be a sequence of musical bars. Each bar contains a set of notes where the number of notes can be different from one bar to the next. One way to define a relation $\equiv$ between two bars b_i and b_j is to consider the number of common notes between b_i and b_j. For example, b_i and b_j can be considered in relation if they share at least a certain proportion of their notes, that is to say:

$$b_i \equiv b_j \;\Leftrightarrow\; \frac{2|b_i \cap b_j|}{|b_i| + |b_j|} \geq \lambda,$$

where $|b_i|$ is equal to the number of notes of the bar b_i and $0 \leq \lambda \leq 1$.

With this definition, we compute the hierarchical segmentations of the song as in the previous section, *In My Life* from The Beatles, by choosing $\lambda = 0.5$, which ensures that two bars are equivalent if they share at least half the notes. The results, represented in Figure 5.10, are similar to Figure 5.9 with three main sections of equal length for the second line which correspond to the `Verse/Bridge` section. The `Intro`, `Half-intro` and `Outro` are also represented on this line. Note that some bars contain more than one chord, e.g. in the `Verse` and `Outro`, which is why there are more chords in Figure 5.9 than bars in Figure 5.10.

It is possible to change the relation $\equiv$ between two bars b_i and b_j by adapting the λ threshold. For example, to allow greater flexibility for two bars to be equivalent, we can choose $\lambda = 0.3$, which means that two bars are equivalent if

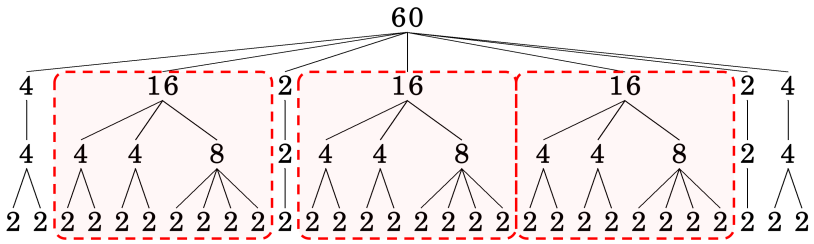
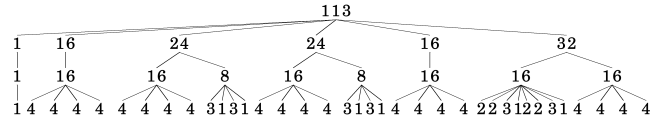


Figure 5.10: Hierarchical segmentations of *In My Life* from The Beatles based on a sequence of bars where the three main sections framed in dotted red are correctly discovered.

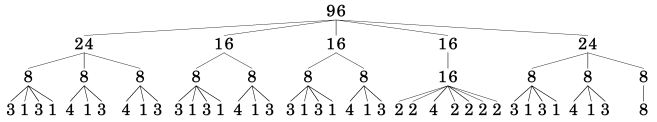
they share at least 30% of common note. We used this value to discover the hierarchical segmentations of some of Chopin's Mazurkas, making it possible to identify bars as similar even with musical variations (such as the one at the end of Mazurka 6-2). Chopin composed Mazurkas throughout his career (with a total of at least 59 Mazurkas), and Witkowska-Zaremba explains that his compositional style has evolved over time [Witkowska-Zaremba 2000]. The first Mazurkas he composed, Op.6, 7, 17 and 24, have a simple segmentation and make extensive use of repetition. A ground truth is given by Witkowska-Zaremba on the segmentation of seven of these Mazurkas: 6-1, 6-2, 7-1, 7-2, 17-3, 24-1, 24-3, which is displayed in Figure 5.11. Other Mazurkas composed later have a more complicated segmentation. These seven Mazurkas are therefore a good study material for academic purposes and for applying our algorithm developed in this chapter. The results are displayed in Figure 5.11 with a tree representation and are commented in Table 5.1. When the Mazurka includes an anacrusis (i.e. a note or set of notes, which can be considered as an incomplete bar, that precedes the first bar of a piece) this is interpreted as a bar by our algorithm. In the case of Mazurkas, this is usually just one note, which is correctly discovered by the algorithm and indicated by a "1" on the far left in the tree representation in Figure 5.11 for Mazurkas 6-1, 7-2, 17-3 and 24-1. However, it is important to point out that even though the results are good overall, these seven mazurkas have a very simple structure that can be detected based essentially on repetition. We believe that this method would not work as well for discovering the segmentations of Chopin's other Mazurkas.

Finally, it is also possible to define the relation between two bars without considering the number of common notes. For instance, by determining the musical chord or key of a bar using a key-finding function, Key, which can be based on the *Krumhansl-Schmuckler Key-Finding Algorithm* [Temperley 1999] or the *Spiral Array Model* [Chew 2001]. Then, we can define the Key relation $\equiv_{key}$ between two bars b_i and b_j by:

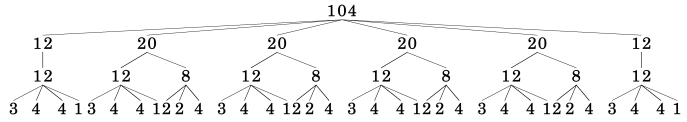
$$b_i \equiv_{key} b_j \;\Leftrightarrow\; \text{Key}(b_i) = \text{Key}(b_j)$$



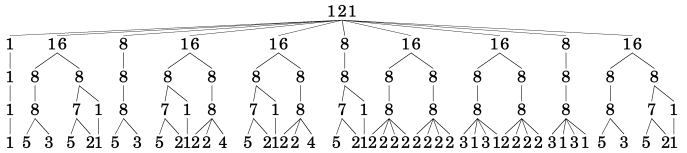
(a) Mazurka 6-1 (a + $A_{16}A_{16}B_8A_{16}B_8A_{16}C_{16}A_{16}$).



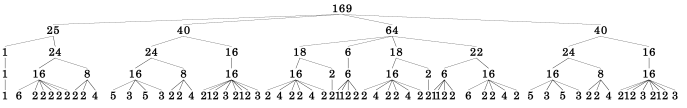
(b) Mazurka 6-2 ($i_8A_8A_8B_8A_8B_8A_8C_{16}i_8A_8A'_8$).



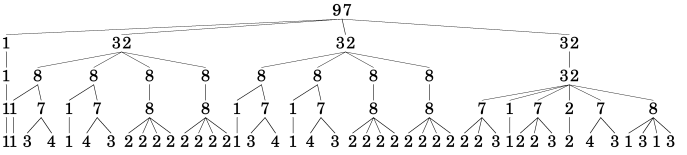
(c) Mazurka 7-1 ($A_{12}A_{12}B_8A_{12}B_8A_{12}C_8A_{12}$).



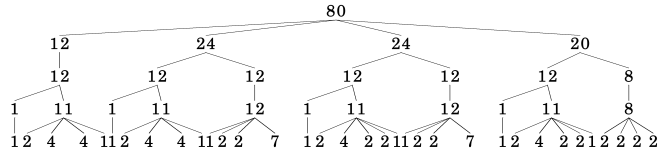
(d) Mazurka 7-2 (a + $A_{16}A_{16}B_8A'_8B_8A'_8C_8D_8C_8D_8C_8A_{16}$).



(e) Mazurka 17-3 (a + $A_{16}B_8A_{16}B_8A_{16}C_{16}D_8C_{16}D_8C_{16}A_{16}B_8A_{16}$).



(f) Mazurka 24-1 (a + $A_{16}B_{16}A_{16}B_{16}C_{16}A_{16}$).



(g) Mazurka 24-3 ($A_{12}A_{12}B_{12}A_{12}B_{12}A_{12}K_8$).

Figure 5.11: Results of the algorithm visualized with a tree representation on seven of Chopin's Mazurkas. The ground truth is shown in parentheses, where "a +" indicates an anacrusis.

Table 5.1: Commented algorithm results on Chopin's Mazurkas.

M	Results
6-1	Good results on the third line of the tree. The anacrusis is isolated. The $A_{16}B_8$ sections are interpreted as a large section on the second line because they are repeated.
6-2	Good results on the third line of the tree. The $i_8A_8A_8$ and B_8A_8 sections are interpreted as a large section on the second line because they are repeated (the repetition is detected even with the A'_8 variation).
7-1	Good results on the third line of the tree. The $A_{12}B_8$ sections are interpreted as a large section on the second line because they are repeated.
7-2	Acceptable results on the third line of the tree, but the A_{16} section is divided into two parts. The anacrusis is isolated. The second line merges the second half of A_{16} with B_8.
17-3	Acceptable results on the fourth line of the tree. The anacrusis is isolated. The sections $A_{16}B_8A_{16}$ (respectively $A_{16}B_8$) are interpreted as a large section on the second line (respectively third line) because they are repeated.
24-1	Acceptable results. The anacrusis is isolated. Sections A_{16} and B_{16} are divided into two because they are composed of two identical parts. Sections $A_{16}B_{16}$ are interpreted as a large section on the second line because they are repeated.
24-3	Good results on the third line of the tree. The $A_{12}B_{12}$ sections are interpreted as a large section on the second line because they are repeated.

5.5 Conclusion

In this chapter, we have further expanded the generalization of *Correlative Matrices* that began with their parametric extension to accept a wide family of relations that define the degree of approximation of repeated patterns. In order to avoid overlapping patterns, we have proposed the *Adapted Correlative Matrix*, a data structure which represents the repeated patterns of a musical sequence without overlaps. We then defined the novel *distinctiveness* criterion, which characterizes the number of repeated patterns that starts at the same time or at the end of a pattern, and proposed an algorithm that iteratively selects patterns based on their distinctiveness to generate hierarchical segmentations. We have demonstrated the usefulness of the proposed method with various music objects from symbolic music representations. Several such relations that are musically relevant have also been presented, but many more can be conceived, including relations that would apply to audio-based objects, for instance, between two frames of a spectrogram. By adjusting which relation to focus on, it is possible to access a much broader meaning of what constitutes a repeated pattern, allowing this approach to be applied to music

genres which do not typically exhibit the strong repetitions that are usually required, while preserving the algorithm lightweight advantage.

However, it is important to mention that the proposed method in this chapter has some limitations. First of all, since the algorithm iteratively detects segmentations, a poor choice at the early stages of the algorithm causes a boundary that remains unchanged for the others discovered segmentations. Secondly, this algorithm is based exclusively on repetition to obtain hierarchical segmentations, but other criteria must also be taken into account to discover musical sections, such as homogeneity, as covered in Chapter 4. Lastly, we have not tested this method on a large database to evaluate it properly which would provide a better understanding of its weaknesses.

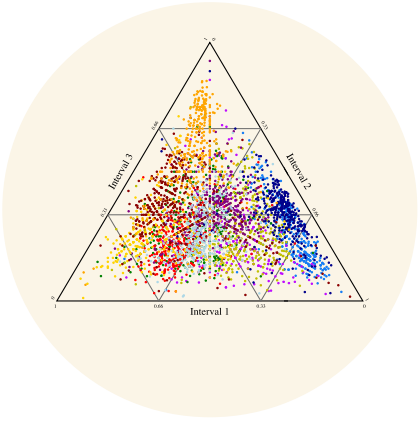
Finally, the representation and algorithms proposed in this chapter can be easily adapted to large datasets without requiring data training. However, at the moment, human knowledge or feedback are still very important in order to pick a representation and relation that is appropriate for the piece and the desired outcome. We expect that the Adapted Correlative Matrix representation would be a powerful tool for machine learning methods that can tailor relation operators to specific music input and find explanatory models for music segmentation.

Part III

Musical Performance

Chapter 6

Characterizing and Interpreting Music Expressivity Through Rhythm and Loudness Simplices



Contents

This chapter presents a novel approach to represent the expressivity of recorded performances using a 2-simplex, a graphical representation used to visualize three-interval rhythms. We analyze the MazurkaBL dataset, which contains beat-level tempo and loudness data of over 2000 recorded performances of 46 Chopin Mazurkas. Mazurkas' triple time lends themselves well to the 2-simplex; the expressive features of each three-beat bar map directly to unique points in the 2-simplex.

Section 6.1 reviews the existing computational methods for music performance analysis. Section 6.2 introduces the rhythm simplex and reviews related work. Section 6.3 presents the dataset we used and explains why the 2-simplex representation is relevant to this dataset. Section 6.4 describes the proposed method, by first providing a transformation to convert a three-interval information into the 2-simplex, then proving that the choices of inter-beat intervals or tempo data are equivalent when timing variation are small, and explaining the impact of smoothing the data in the 2-simplex representation. Section 6.5 presents analyses and interpretations that can be made with the simplices, such as visualizing time suspensions, characterizing the notion of regularity of a performance, and identifying the bars with notable expressive variations. Section 6.6 provides other applications for the 2-simplex representation, and Section 6.7 concludes this chapter.

This chapter is an extended version of the article "Characterizing and Interpreting Music Expressivity through Rhythm and Loudness Simplices" [Lascabettes 2023].

6.1 Computational Models for Music Performances

Characterizing and interpreting expressivity in performed music is a fundamental problem in fields such as musicology, music perception, and music analysis. Moreover, the emergence of computational models has produced significant advances in expressive music research [Widmer 2004]. Computational modeling of expressive music performance requires large-scale databases; to create such databases, it is more expedient to focus on piano music. We focus on the piano pieces of the romantic period, which allow for greater expressive variations. A significant amount of this repertoire has been written for solo instruments, which makes comparative analysis of performances more straightforward. Furthermore, with the existence

of computer-controlled pianos such as the Bösendorfer Enspire PRO and Steinway Spirio, realistic piano performances can be readily captured with accuracy, which is not the case with other instruments. Therefore, the majority of performance research focuses on piano music.

Due to the resources amassed for and made available by the Mazurka Project[1], numerous studies have been based on Chopin's Mazurkas. A main purpose of the initial studies was to identify correlations between performed tempo and between performed loudness features [Cook 2007, Sapp 2007, Sapp 2011, Cook 2014]. These analyses were based on selected Mazurka recordings. More recently, beat level tempo and loudness information of 2000 recorded Mazurkas have been extracted and made available for research [Kosta 2018]. Other studies have focused on generating Mazurka performances using machine learning models [Shi 2021]. The two main expressive parameters considered are tempo and loudness variations [Cancino-Chacón 2018]. Using these two parameters, trajectories in tempo-loudness space [Langner 2003] were traced to represent the performance. Another representation of tempo and loudness curves for analysis is the arc model [Gabrielsson 1987, Todd 1992]. Arc models have been successfully used to determine the segmentation of a piece based only on the loudness and tempo curves versus time [Widmer 2003, Chuan 2007, Stowell 2013].

In this chapter, we propose to represent beat-level tempo and loudness information of a Mazurka performance in tempo and loudness simplices, a graphical representation used in music to analyze the perception of three-interval rhythms. This is the first time these simplices are applied to visualize Mazurka performances. This representation makes it straightforward to determine performance characteristics such as the accented beat in a bar or the regularity of a performance.

6.2 The Rhythm Simplex

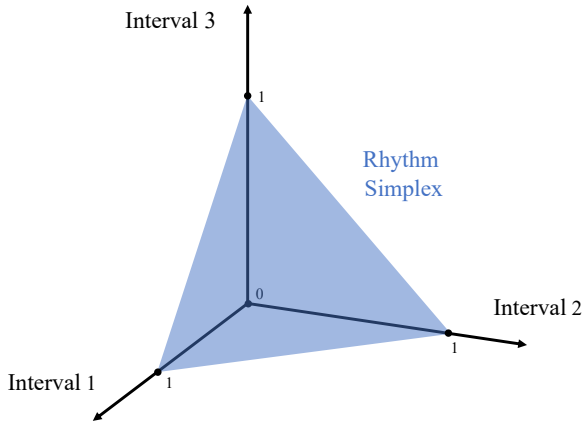
In this section, we introduce the rhythm simplex, a graphical representation used to visualize three-interval rhythms, and related work.

6.2.1 Presentation of the Rhythm Simplex

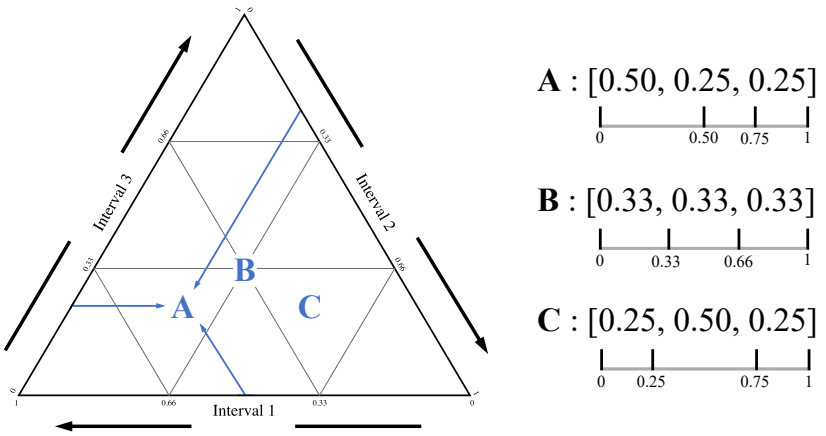
The *rhythm simplex*, also known as a *rhythm chart* or *chronotopological map*, is a graphical representation developed for visualizing three-interval rhythms. The idea of the rhythm simplex is that any three-interval rhythm can be represented in a 2-dimensional plot if the total duration of the rhythm is fixed (usually normalized to 1). For example, $A = [0.50, 0.25, 0.25]$ and $B = [0.33, 0.33, 0.33]$. By fixing the total duration, it is sufficient to know the first two intervals to deduce the third one. So a 2-dimensional plot can be used to visualize the rhythm. Thus, a three-interval rhythm is mapped to a unique point in the rhythm simplex. As illustrated in Figure 6.1(a), this 2-dimensional plot is a 2-simplex, i.e. a triangle. Figure 6.1(b)

[1] http://www.mazurka.org.uk/

represents the rhythm simplex where each side of the triangle corresponds to one of the three intervals. In this figure, each three-interval rhythm is represented by a unique point. The rhythm A is located by the light blue arrows, while the rhythm B is in the middle of the rhythm simplex.



(a) The space of three-interval rhythms in the 2-simplex.



(b) Rhythm simplex where the three-interval rhythms A, B, and C are represented by unique points in the 2-simplex.

Figure 6.1: Three-interval rhythms in the rhythm simplex (adapted from Desain & Honing [Desain 2003]).

6.2.2 Previous Works Related to the Rhythm Simplex

The use of the simplex representation in music was introduced by Desain and Honing [Honing 2002, Desain 2003] to understand how listeners perceive rhythm categories. They asked the listeners to identify three-interval rhythms on a continuous scale to determine areas in the rhythm simplex representing equivalence classes. These rhythm equivalence classes can evolve according to parameters such

as tempo [Handel 1993], loudness or melodic structure [Tekman 1997]. Vaquero and Honing [Patricio 2014] created paths within these areas to generate performances. Based on a type of formal grammar called Lindenmayer systems [Lindenmayer 1968], they defined paths in the rhythm simplex to generated expressivity in music. Bååth et al. [Bååth 2014] implemented a dynamical systems model to reproduce the categorical choices of listeners to retrieve these areas in the rhythm simplex. More recently, Jacoby and McDermott [Jacoby 2017] ran a study where random rhythms in the rhythm simplex had to be reproduced by participants. Participants converged to rhythms having integer ratios after five iterations. They showed that the areas defined in [Desain 2003] have little dependence on musical training but are highly dependent on cultural biases. For instance, listeners in the United States had different results than native Amazonian listeners. Nave et al. [Nave 2021] conducted a similar experiment to the one in [Jacoby 2017] with iterative reproductive tasks based on random rhythms in the rhythm simplex, but with children. They demonstrated the existence of rhythm priors in children, also related to cultural biases, which would develop in middle childhood.

Finally, these different studies used the rhythm simplex to understand the perception of three-interval rhythms, whether in populations of different cultures or of different ages. Only Vaquero and Honing have applied it to musical performances, but for generation rather than analysis. Here, we propose a new approach to use the rhythm simplex to represent musical performances in order to characterize and interpret music expressivity.

6.3 Mazurka Performances

In this section, we present the dataset that we use in this chapter and explain how the rhythm simplex can be used to represent the essential expressive features of Mazurka performances.

6.3.1 The Dataset

To map a large number of Mazurka performances to the rhythm simplex for analysis, we used the MazurkaBL dataset [Kosta 2018]. This is currently by far the largest database of annotated performed classical music having multiple performances of each piece. The MazurkaBL contains beat level duration and loudness annotations for over 2000 recorded performances of 46 Chopin Mazurkas. There are, on average, more than 40 distinct performances per Mazurka.

This dataset was made by manually annotating the beats of one recording of a Mazurka and automatically transferring these annotations to other recordings of the same piece through audio alignment. It is important to note that we used a smoothed version of the loudness data. Kosta et al. [Kosta 2018] filtered the loudness annotations by local regression using a weighted linear least squares and a 2nd degree polynomial model (the LOESS method of MATLAB's smooth function)

with window sizes that are 1/30-th of the length of the recorded Mazurka while the tempo annotations are raw. It is therefore essential to take this into account, as it has an impact on the position of points in the 2-simplex and therefore on their interpretation, as explained in the following sections.

6.3.2 Representing Mazurka Performances in 2-Simplices

As described in Section 6.2.1, the rhythm simplex represents a three-interval rhythm as a unique point in a 2-dimensional plot. The use of the rhythm 2-simplex must therefore be applied to rhythm data that can be split into groups of three intervals, which is not the case for any general temporal data. However, Chopin's Mazurkas is particularly well suited to rhythm simplex representation because Mazurkas are folk dances mostly in triple meter, i.e., they are mostly pieces with three beats in each bar. On very rare occasions, some of Chopin's Mazurkas may contain bars that are not in triple meter, in which case these bars are removed from the analysis. Otherwise, the three beats can be seen as three intervals which have equal duration in the score but whose time is rendered differently in performance. Therefore, a bar of a performed Mazurka can be mapped into the rhythm simplex as a single point. By viewing the recording of a Mazurka as a sequence of performed three-beat bars, we can represent the recording by a set of points in the rhythm simplex. The rhythm simplex can show not only the timing variations in each bar, it can also display the beat accentuation because the loudness data of each beat of a three-beat bar can also be represented in a 2-simplex. For Mazurkas beginning with an anacrusis, we ignore the notes that come before the first full bar of music. Therefore, the essential expressive features of the musical performance of the Mazurkas can be visualized using 2-simplices because most of bars have three beats.

6.4 The Method to Represent Mazurka Performances in 2-Simplices

In this section, we give the transformation that we use to convert three-interval information into points in the 2-simplex. For the information related to rhythm, we prove that using inter-beat intervals or using tempo are equivalent in the 2-simplex representation. For the information related to loudness, we introduce the loudness simplex and explain how smoothing the data impacts the coordinates in the simplex.

6.4.1 Computing Points in the 2-Simplex

Previous articles on the rhythm simplex (see Section 6.2.2) have not explicitly provided the transformation to map a three-interval rhythm to a point in the rhythm simplex. Here, we provide the transformation we use to convert three-interval information to a point in 2-simplex.

First, we fix the vertices of the 2-simplex on the unit circle, i.e. the vertices are $(0,1)$, $(\frac{\sqrt{3}}{2}, -\frac{1}{2})$, and $(-\frac{\sqrt{3}}{2}, -\frac{1}{2})$ (which are represented in Figure 6.1(b) by the top vertex, the bottom right vertex, and the bottom left vertex, respectively). In this case, the middle of the triangle is at the origin $(0,0)$ (represented by point B in Figure 6.1(b)).

Let $B = (b_1, b_2, b_3)$ represent some properties of a three-beat bar such that $b_1 + b_2 + b_3 = 1$, where b_1, b_2, and b_3 are three positive numbers. The corresponding point (x, y) in the 2-simplex is defined by:

$$x = \frac{\sqrt{3}}{2} - \frac{\sqrt{3}}{2} b_3 - \sqrt{3} b_1 \tag{6.1}$$

$$y = \frac{3}{2} b_3 - \frac{1}{2} \tag{6.2}$$

Reciprocally, given a point (x, y) in the 2-simplex, the corresponding bar $B = (b_1, b_2, b_3)$ is defined by:

$$b_1 = \frac{1}{3} - \frac{1}{\sqrt{3}} x - \frac{1}{3} y \tag{6.3}$$

$$b_2 = \frac{1}{3} + \frac{1}{\sqrt{3}} x - \frac{1}{3} y \tag{6.4}$$

$$b_3 = \frac{1}{3} + \frac{2}{3} y \tag{6.5}$$

This defines a bijection between the 2-simplex and a normalized feature of the three-beat bars.

6.4.2 Inter-Beat Intervals and Tempo Data in the 2-Simplex

To represent a recorded Mazurka performance (or any performed piece in triple time) in the rhythm simplex, we compute inter-beat intervals from the beat onset information in the MazurkaBL. This gives us the duration of each beat, (d_1, d_2, d_3), in a bar. We scale this vector so that the elements sum to one to get (b_1, b_2, b_3), where $b_i = \frac{d_i}{d_1 + d_2 + d_3}$ for $i = 1, 2, 3$. The normalized durations in that bar then map to a point in the rhythm simplex according to Equations 6.1 and 6.2. In this way, we map the performed beat durations of all bars to the rhythm simplex. For example, in Figure 6.2, the different interpretations of the first four bars of Mazurka 6-1, obtained from the MazurkaBL dataset, are represented in the rhythm simplex. In this case, the first note is not considered, as Mazurka 6-1 begins with an anacrusis. Each point in a rhythm simplex corresponds to an interpretation of a bar, where the temporal deformation of the three beats (d_1, d_2, d_3) can be visualized. We can notice that all the performers follow a similar path. For instance, for the first bar, all points tend to be on the left in the simplex (meaning a long first beat and a short second beat). Whereas on the second and fourth bars, the points are on the right in the simplex (meaning a short first beat and a long second beat). Finally, for the

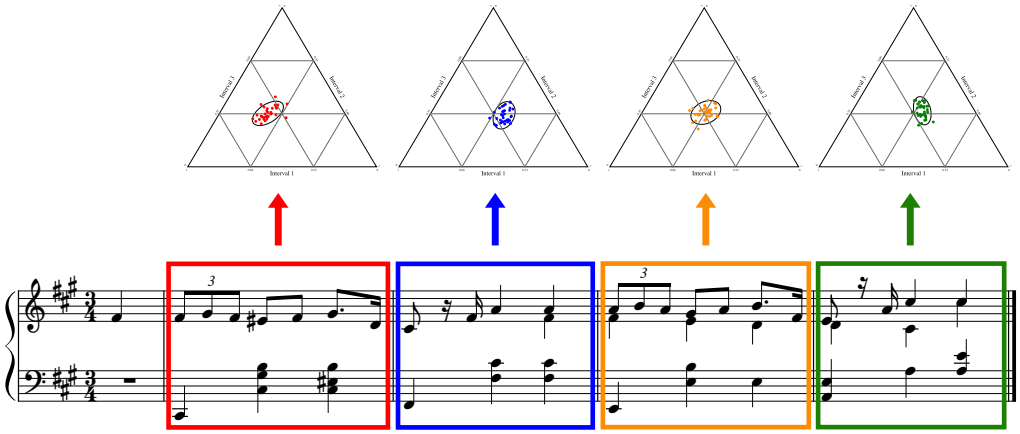


Figure 6.2: Visualization of different interpretations of the first four bars of Mazurka 6-1 in the rhythm simplex. Each point in a rhythm simplex corresponds to the interpretation of a bar by a performer, where the temporal deformation of the three beats of the bar is visualized.

third bar, interpretations are on average more regular as the points are concentrated in the middle of the simplex.

Previous studies on Mazurkas use tempo data rather than duration data. To generate comparable results, we can also compute the respective beat-to-beat tempo of a bar (t_1, t_2, t_3) with $t_i = 60/d_i$ for $i = 1, 2, 3$, and normalize the values for mapping to a 2-simplex. However, which data should we choose: inter-beat intervals or tempo? We show with the following proposition that the points in the 2-simplex based on inter-beat intervals or tempo data have approximately the same coordinates up to one symmetry with respect to the origin. This proposition shows that the choice of tempo or duration does not matter in the 2-simplex representation.

Proposition 6.1: Inter-beat intervals or tempo?

Let (d_1, d_2, d_3) denote three inter-beat intervals of a bar and (x, y) the corresponding point in the 2-simplex. When (x, y) is close to the origin, the 2-simplex mapping of the beat-to-beat tempo of the same bar (t_1, t_2, t_3) is approximately $(-x, -y)$.

Proof. Let $d = d_1 + d_2 + d_3$ and $b_i = \frac{d_i}{d}$ for $i = 1, 2, 3$. Now, normalize the vector (t_1, t_2, t_3) and find its coordinates (x_T, y_T) in the 2-simplex.

$$\begin{aligned}
\frac{1}{t_1+t_2+t_3}(t_1,t_2,t_3) &= \frac{1}{\frac{60}{d_1}+\frac{60}{d_2}+\frac{60}{d_3}}(\frac{60}{d_1},\frac{60}{d_2},\frac{60}{d_3}) \\
&= \frac{1}{\frac{1}{d_1}+\frac{1}{d_2}+\frac{1}{d_3}}(\frac{1}{d_1},\frac{1}{d_2},\frac{1}{d_3}) \\
&= \frac{d}{d}\frac{1}{\frac{1}{d_1}+\frac{1}{d_2}+\frac{1}{d_3}}(\frac{1}{d_1},\frac{1}{d_2},\frac{1}{d_3}) \\
&= \frac{1}{\frac{d}{d_1}+\frac{d}{d_2}+\frac{d}{d_3}}(\frac{d}{d_1},\frac{d}{d_2},\frac{d}{d_3}) \\
&= \frac{1}{\frac{1}{b_1}+\frac{1}{b_2}+\frac{1}{b_3}}(\frac{1}{b_1},\frac{1}{b_2},\frac{1}{b_3}) \\
&= \frac{1}{b_2b_3+b_1b_3+b_1b_2}(b_2b_3,b_1b_3,b_1b_2)
\end{aligned}$$

with (b_1, b_2, b_3) expressed with x and y as shown in Equations 6.3, 6.4 and 6.5. The vector (d_1, d_2, d_3) represents the duration of each beat of a bar. In a performance without large timing variations, the b_i's are close to $\frac{1}{3}$ and (x, y) is close to $(0, 0)$ in the 2-simplex (i.e. close to point B shown in Figure 6.1(b)). When x and y are close to zero, x^2, y^2, and xy are negligible. In this case, we get:

$$b_1b_2 \approx \frac{1}{9}(-2y+1), \qquad b_1b_3 \approx \frac{1}{9}(y-\sqrt{3}x+1),$$

$$b_2b_3 \approx \frac{1}{9}(y+\sqrt{3}x+1), \qquad b_2b_3+b_1b_3+b_1b_2 \approx \frac{1}{3}$$

Applying the transformation in Equations 6.1 and 6.2 to the vector (t_1, t_2, t_3) normalized with the approximations (i.e. x^2, y^2, and xy are negligible), we have:

$$x_T \approx \frac{\sqrt{3}}{2} - \frac{\sqrt{3}}{2}(\frac{1}{3}(-2y+1)) - \sqrt{3}(\frac{1}{3}(y+\sqrt{3}x+1)) = -x,$$

and

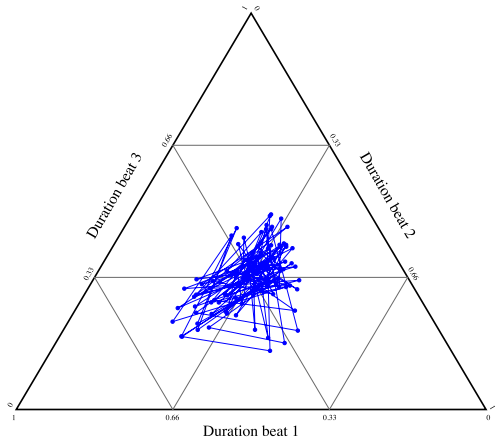
$$y_T \approx \frac{3}{2}(\frac{1}{3}(-2y+1)) - \frac{1}{2} = -y$$

□

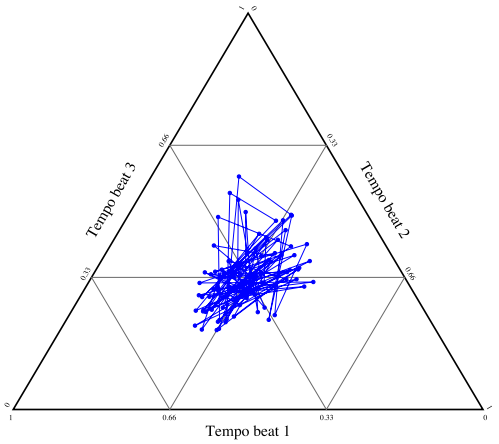
To illustrate this result, Mazurka 6-1 performed by Luisada is represented in a 2-simplex in Figure 6.3(a) based on inter-beat intervals, and in Figure 6.3(b) based on beat-to-beat tempo. We can see that the point sets are similar in both figures up to one symmetry with respect to the origin.

6.4.3 The Loudness Data in the 2-Simplex

Because loudness data are also available in the MazurkaBL dataset, we extend the rhythm simplex to the representation of loudness. We define the *loudness simplex* by considering the relative proportion of loudness for each beat in a three-beat



(a) Representation of Mazurka 6-1 performed by Luisada in the 2-simplex based on beat-to-beat duration.



(b) Representation of Mazurka 6-1 performed by Luisada in the 2-simplex based on inter-beat intervals.

Figure 6.3: Inter-beat intervals and tempo data represented in a 2-simplex to illustrate the symmetry with respect to the origin between the two sets of points.

bar. This is exactly the same idea as for rhythm; as we shall see later, it provides additional insights.

Because the loudness data have been smoothed with a LOESS filter by Kosta et al. [Kosta 2018], the loudness curves as a function of time tend to be smoother, involving fewer local maxima or minima. This has a direct consequence on the coordinates of the points in the loudness simplex. Figure 6.4 shows the triangle of the 2-simplex split into four regions: two areas, in blue, located at the top right and bottom left of the simplex; and, two areas, in red, located at the top left and bottom right. The two blue areas correspond to points in the simplex where the loudness increases or decreases with time within a bar. At the top right of the simplex, the elements are increasing, while at the bottom left they are decreasing. Musically, these areas translate to bars where the performer has an ascending (top right) or descending (bottom left) movement within that expressive property. Conversely, the points in the simplex are located in the red zone if the second beat is a local minimum or maximum, which is more seldom seen with loudness data because the data have been smoothed. Thus, as can be seen in Figures 6.6(a) and 6.6(b), when representing the loudness data in the 2-simplex, the points tend to stay in the blue area because the data have been smoothed. However, this does not exclude performance information as we will see in the rest of this chapter. Note that this reasoning applies to any type of smoothed data projected into a 2-simplex.

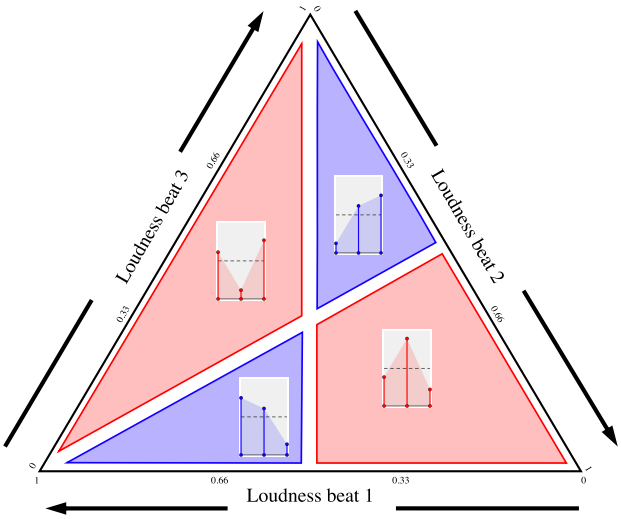


Figure 6.4: Area in the loudness simplex characterized by an increasing loudness curve (in blue) or local maxima or minima (in red) in the bar.

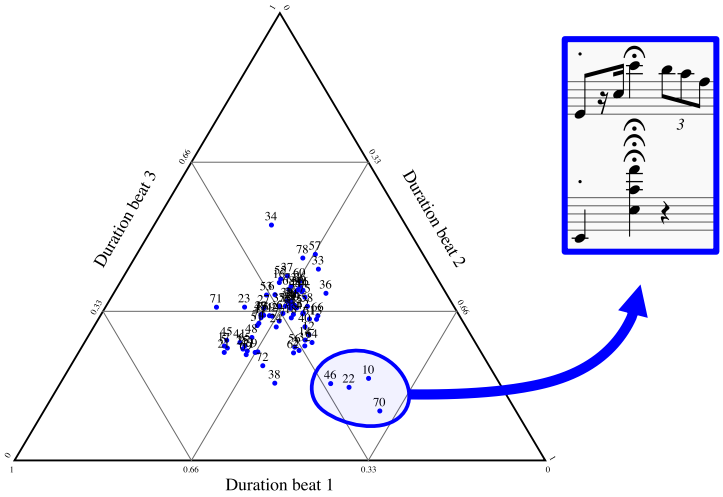
6.5 Analysis and Interpretation of Musical Expressivity Using the Simplices Representation

In this section, we present analyses and interpretations that can be derived from the simplices representation. This includes visualizing tipping points (here realized as time suspensions) introduced by the performer, defining the regularity of a performance, and identifying bars that exhibit notable expressivity.

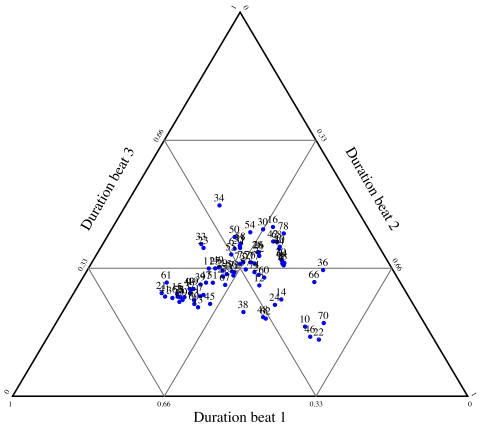
6.5.1 Visualize Time Suspensions in the Simplex

Some recorded Mazurkas display musical tipping points [Chew 2016] which present as significant temporal suspensions. These time elongations may be indicated in the score as fermatas or may simply be inserted by the performer. Representing these Mazurka performances in the rhythm simplex allows these temporal deformations to be visible. For example, in Mazurka 24-3, bars 10, 22, 46, and 70 are identical and have a fermata marked and executed on the second beat. Thus, the duration of the second beat of these bars is significantly longer than others, making them more important. Therefore, the points corresponding to these bars are situated at the bottom right of the rhythm simplex. For example, by representing Uninsky's interpretation of Mazurka 24-3 in the rhythm simplex, we can easily extract these bars as shown in Figure 6.5(a). These temporal suspensions are present in most interpretations of Mazurka 24-3, so these four points in the bottom right of the simplex are seen in other interpretations as well. For example, they are also found in the Biret and Kushner interpretations shown in the rhythm simplex in Figures 6.5(b) and 6.5(c). These tipping points [Chew 2016] allow performers to pause to highlight

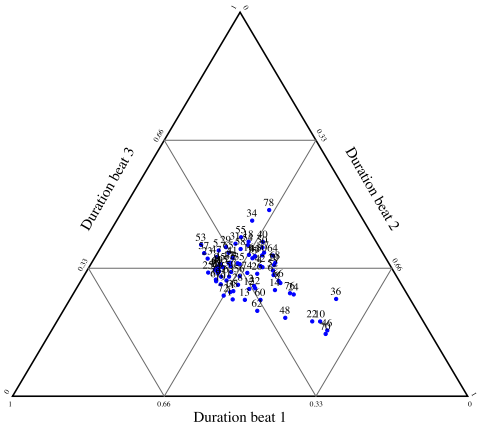
certain notes or harmonies to create tension and anticipation [Herremans 2016]; in the case of Figure 6.5, it is the highest note of the bar on the second beat.



(a) Uninsky's interpretation of Mazurka 24-3.



(b) Biret's interpretation of Mazurka 24-3.



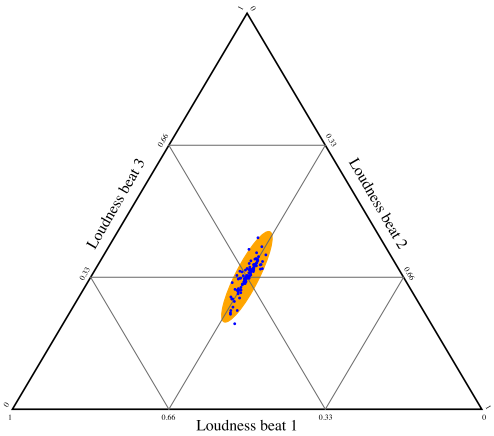
(c) Kushner's interpretation of Mazurka 24-3.

Figure 6.5: Temporal suspensions of the second beat of bar 10, 22, 46, and 70 of Mazurka 24-3 indicated as fermatas in the score and visualized by isolated points in the bottom right of the rhythm simplex.

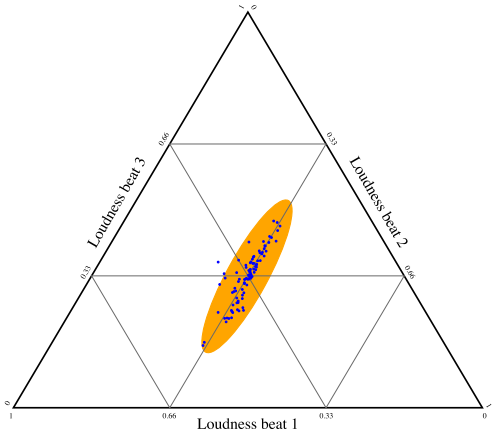
6.5.2 Characterizing the Regularity of a Performance Using the Simplices Representation

During a musical performance, some performers allow themselves greater latitude in their temporal and loudness variations than others. However, quantifying these variations to define the regularity of a performance is not easy [Cancino-Chacón 2018].

If a performer chooses to always accentuate the first beat of each bar, the performance could be described as regular, even though the differences in tempo and volume per beat can be large during the piece. Therefore, it is not sufficient to characterize the regularity of a performance by the sum of the deviations per beat from the average of tempo and loudness. Hence, we propose to quantify the regularity of a performance using the 2-simplices. To get an intuition about this, see Figure 6.6(a) where the points are very dense in the loudness simplex meaning that the loudness variations are very regular. On the other hand, in Figure 6.6(b), the points are more spread out because the loudness varies less regularly.



(a) Small area of the ellipse in the loudness simplex from Kapell's interpretation of Mazurka 6-2 indicating an interpretation with regular loudness.



(b) Large area of the ellipse in the loudness simplex from Ohlsson's interpretation of Mazurka 6-2 indicating an interpretation with non-regular loudness.

Figure 6.6: Example of regular and non-regular performed loudness shown by a small and a large ellipse area in the loudness simplex. Kapell's ellipse area is three times smaller than Ohlsson's, pointing to greater loudness regularity.

We propose to define the regularity of a performance from the covariance matrix of the points in the 2-simplex as follows. Let $(x_i, y_i)_{1\leq i\leq n}$ denote the points in the simplex and Σ their covariance matrix:

$$\Sigma = \begin{pmatrix} \mathrm{Cov}(X,X) & \mathrm{Cov}(X,Y) \\ \mathrm{Cov}(Y,X) & \mathrm{Cov}(Y,Y) \end{pmatrix},$$

where $X = (x_i)_{1\leq i\leq n}$, $Y = (y_i)_{1\leq i\leq n}$,

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \quad \text{and} \quad \mathrm{Cov}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})$$

Let λ_1 and λ_2 denote the eigenvalues of Σ. They define an ellipse[2], whose semi-axes lengths are $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$. The orientation is derived from Σ as well, but not involved

[2]We used the matplotlib package https://matplotlib.org/stable/gallery/statistics/confidence_ellipse.html, with the default setting n_std=3.0, for the ellipses in Figure 6.6.

in the regularity computation. We define the regularity value of a performance as the inverse of the area of that ellipse.

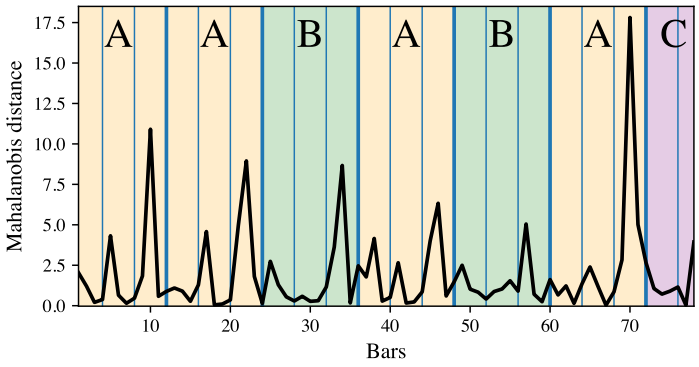
Since the area of an ellipse is equal to the multiplication of π by the length of the semi-major and the semi-minor axes, it is easy to compare the regularity of two performances in the 2-simplex. Using the example in Figure 5, Kappel (λ_1 and λ_2 as eigenvalues) is 2.995 times more regular than Ohlsson (λ'_1 and λ'_2 as eigenvalues) because $\sqrt{\lambda'_1\lambda'_2}/\sqrt{\lambda_1\lambda_2} = 2.995$. This reasoning can be applied across interpretations to other expressive features like beat-level tempo and duration.

6.5.3 The Most Distant Points in the Simplex Indicate Bars With Notable Musical Expressivity

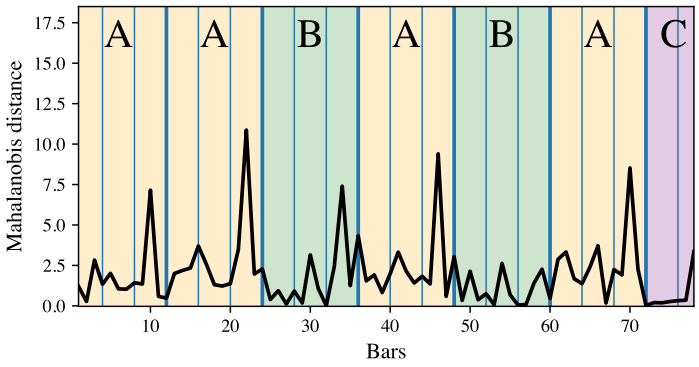
The ellipses described in the previous section correspond to points having the same Mahalanobis distance to the center [Mahalanobis 1936]. Distinct from Euclidean distance, the Mahalanobis distance takes into account the distribution of points in the simplex. Given two points (x_1, y_1) and (x_2, y_2) in the 2-simplex, the Mahalanobis distance is defined by:

$$d\left(\begin{pmatrix}x_1\\y_1\end{pmatrix},\begin{pmatrix}x_2\\y_2\end{pmatrix}\right) = \sqrt{\begin{pmatrix}x_1-x_2\\y_1-y_2\end{pmatrix}^T \Sigma^{-1} \begin{pmatrix}x_1-x_2\\y_1-y_2\end{pmatrix}}$$

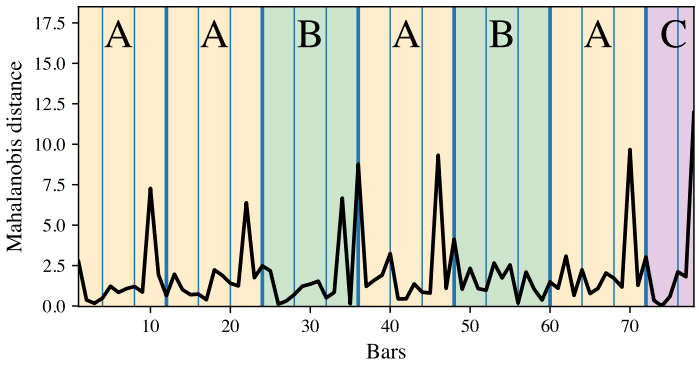
This distance is useful in our case because there is often a correlation between the coordinates of the points in the simplex, in particular, for points in the loudness simplex, the x coordinate increases when the y coordinate increases (but the reasoning is also correct for the rhythm simplex). For example, in Figure 6.6, the points are aligned in the simplex because the loudness data have been smoothed, as detailed in Section 6.4.3. In this case, the Mahalanobis distance allows us to give more importance to the points that do not follow this alignment. Thus, the points with a high distance value compared to the average are the bars that present a strongly divergent musical expressivity according to the simplex feature. For instance, we can identify the temporal elasticity in the interpretations of Mazurka 24-3 which are represented in the rhythm simplex in Figure 6.5. We computed the Mahalanobis distance between each simplex point and the mean for these three interpretations and plotted the result in Figure 6.7. Bars that have an extreme time elongation on the second beat, i.e. bars 10, 22, 46, and 70 of Mazurka 24-3 represented by the four points at the bottom right of the rhythm simplex in Figure 6.5, are located almost at the end of each A section and have a high distance value due to their divergent interpretation as compared to other bars, indicated by local maxima in Figure 6.7. Thus, by considering the maximum values of the Mahalanobis distance between the simplex points and their mean, we can detect the bars that have a notable expressive variation. For example, in Figure 6.7, bar 34 also has a high distance value in all three interpretations because the first beat is considerably longer than the other two, which can be seen in the top corner of the rhythm simplices in Figure 6.5.



(a) Uninsky's interpretation of Mazurka 24-3.



(b) Biret's interpretation of Mazurka 24-3.



(c) Kushner's interpretation of Mazurka 24-3.

Figure 6.7: Mahalanobis distance between the points of the interpretations of Mazurka 24-3 in the rhythm simplex shown in Figure 6.5. High distance values are bars with notable expressive variations. For example, bars 10, 22, 46, and 70 have high distance values because their second beats are elongated, while bar 34 has its first beat elongated.

6.6 Other Applications of the Rhythm Simplex

In this section, we present further applications of the rhythmic simplex; we demonstrate that it can also be used to visualize speech-voice rhythms and irregularities in electrocardiograph data.

6.6.1 Application to Speech Voice Rhythms

We employed the rhythm simplex to demonstrate the connection between the rhythm of a spoken sentence and its linguistic content, specifically the words within the sentence. To do this, we conducted research with the Att-HACK database, which contains 20 speakers interpreting 100 French utterances in four different social attitudes with 3-5 repetitions each per attitude [Le Moine 2020]. The four attitudes, which are friendly, seductive, dominant, and distant, make it possible to obtain a variety of pronunciations for each utterance. We therefore assume that the different speakers (male and female), together with the different attitudes repeated several times, cover a wide prosodic range that accounts for real-life pronunciations of those sentences.

To visualize utterances in 2-simplices, we restricted ourselves to utterances of 3 syllables, in order to have three-interval data. There are 11 different utterances from the Att-HACK database that have exactly three syllables. For each of these utterances, there are over 300 different recordings (with different actors and attitudes) for a total of 3972 recordings of utterances with 3 syllables. These utterances and their occurrences are presented in Table 6.1.

Table 6.1: The various three-syllable utterances in the Att-HACK database and their occurrences.

Utterances	Occurrences
par ici	361
au revoir	365
à demain	353
impossible	367
volontiers	358
j'ai toussé	360
peu importe	356
nous partons	364
bonne journée	352
bonsoir jeanne	363
faites leur signe	373

Speech-to-text alignment algorithms provide the duration of each syllable in a recorded utterance [Teytaut 2022]. We have therefore represented the 3972 voice rhythms in the rhythm simplex in Figure 6.8, where each color corresponds to an utterance. This visualization reveals that, despite the differences in pronunciation

that are linked to the speaker (extra-linguistic) and those that are expressive, such as attitude (para-linguistic), each utterance can be found in a certain area of the 2-simplex. That is to say, in the context of this study, where the utterance rhythm can be impacted by different factors (speaker, attitude, linguistic content), we observe that the determining factor is linguistic. The rhythm varies relatively little with fixed linguistic content for different attitudes and different speakers. However, we cannot say that all occurrences of the same linguistic content have the same rhythm either: clusters are not so small and can intersect. For example, the utterance *"par ici"* (represented by blue points) is located to the right of the 2-simplex because the first syllable is always shorter than the other two. On the other hand, the utterance *"faites leur signe"* (represented by orange points) is in the upper corner of the 2-simplex, because the second syllable is always pronounced longer than the other two. This first result shows that the rhythm of the spoken word is strongly correlated with its content. A deeper analysis could be done in future work.

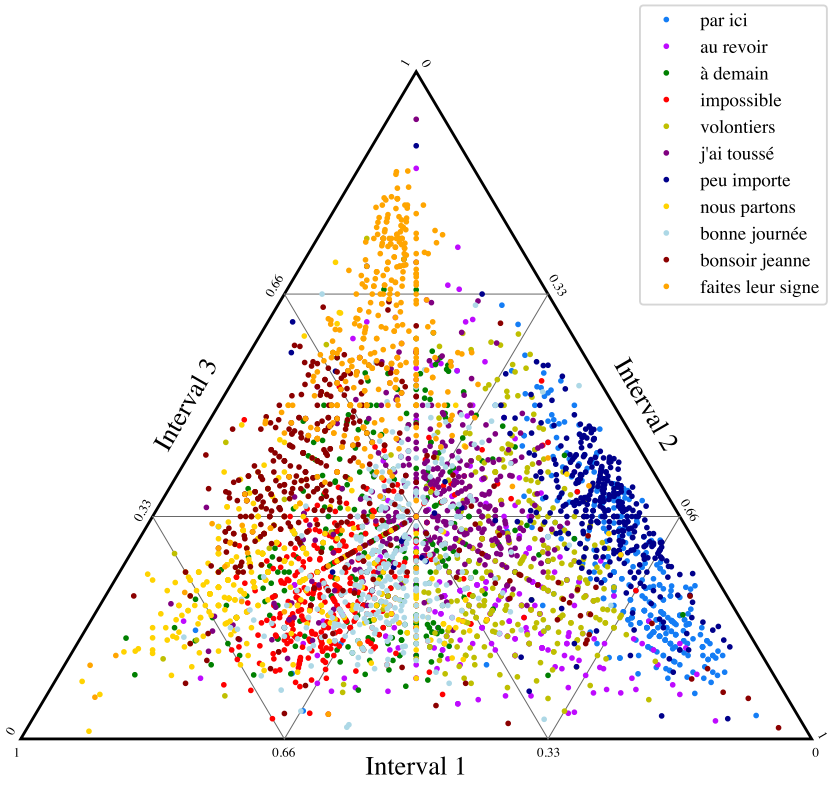


Figure 6.8: Three-syllable utterances from the Att-HACK database represented in the rhythm simplex to visualize that the rhythm of a spoken phrase is related to its content.

6.6.2 Application to Electrocardiograph Data

Finally, we also analyzed electrocardiographic (ECG) data with the rhythm simplex by focusing on RR intervals, which refer to the time intervals between successive R waves on an ECG. They are used to assess heart rate variability and overall cardiovascular health. Previously, integer ratios between two consecutive RR interval have been observed [Chew 2021]. Similarities have been established with musical rhythms, which are also based on integer ratios (at least in Western culture). However, in this study, only two-interval rhythms were studied, so we extend the study to three-interval rhythms, using the rhythm simplex. To do this, we followed the same method as Jacoby and McDermott [Jacoby 2017]. They demonstrated that participants from United States converged on integer ratio rhythms when they reproduced random rhythms. This can be visualized in the rhythm simplex as points condense to rhythms with integer ratios that can be expressed as i-j-k where $1 \leq i, j, k \leq 3$. We therefore analyzed the same ECG data as in Chew's study, which include over 72,000 consecutive RR intervals. To visualize them in the rhythm simplex, we divided the ECG data into series of three consecutive RR intervals with overlaps to obtain 72,000 points in the rhythm simplex. We have displayed these ECG data in the rhythm simplex in Figure 6.9 by adding the rhythms with integer ratios expressed as i-j-k where $1 \leq i, j, k \leq 3$ as in Jacoby and McDermott's study.

First, we can observe that the points do not strongly converge towards rhythms with integer ratios, with the exception of the 1-1-1 rhythm, which signifies a regular heartbeat. However, the points are not randomly distributed in the 2-simplex and seem to be organized in a very particular shape. Note that this shape is not necessarily invariant to a 120-degree rotation. This means that, if there are points in one corner of the 2-simplex, this does not imply that there will also be points in the other corners of the simplex. For example, with the interval series 1-1-3-3-2-1, the corresponding point of the first rhythm 1-1-3 is in the upper corner of the 2-simplex because the third interval is predominant, whereas the other rhythms 1-3-3, 3-3-2, 3-2-1 are not in a corner of the simplex. Therefore, the result obtained in this section demonstrates the difficulty of converting RR intervals into musical rhythms, as we have already tried to do using rhythm simplex [Romero-García 2022b]. However, given the shape obtained in the rhythm simplex from RR intervals, we therefore believe that this representation can be used to characterize specific irregularities in ECG data.

6.7 Conclusion and Future Work

In this chapter, we proposed a novel method to represent performed music using beat duration, tempo and loudness simplices. We analyzed the MazurkaBL dataset using these simplices. MazurkaBL contains more than 2000 recorded performances of 46 of Chopin's Mazurkas with annotations of beat level tempo and loudness values. We provided the equations to map any bounded three-interval information into the

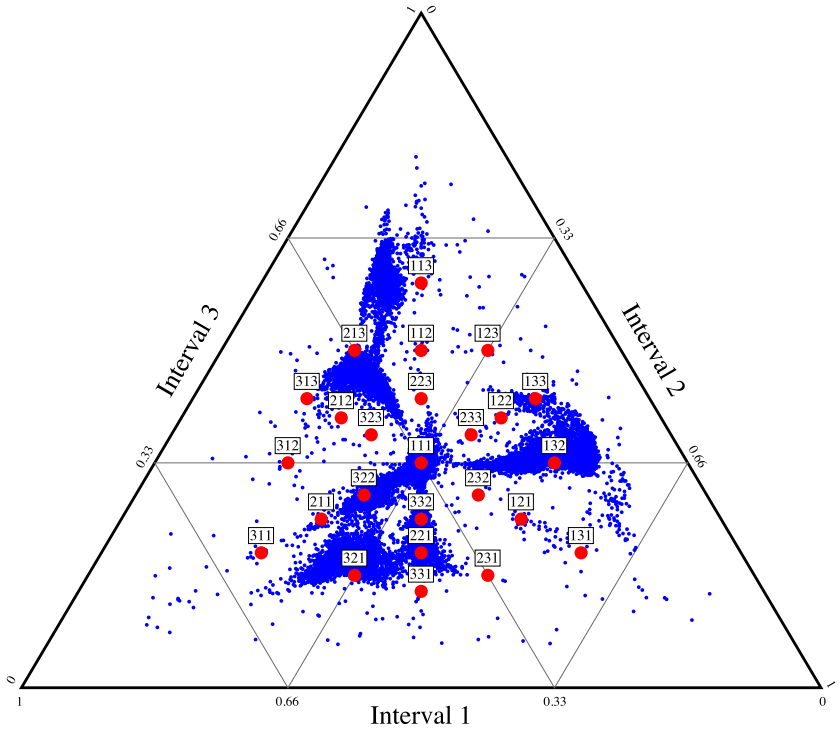


Figure 6.9: ECG data of consecutive RR intervals visualized in rhythm simplex. These data, interpreted as 3-interval rhythms, do not appear to converge towards rhythms with integer ratios.

2-simplex. We proved that the choices of inter-beat intervals or tempo are nearly equivalent in the 2-simplex up to one symmetry with respect to the origin. We also explained the impact of smoothed data in the 2-simplex. Finally, we showed that the simplices facilitate the analysis and interpretation of musical expressivity features. For example, by using the Mahalanobis distance, it is possible to identify bars with notable expressive variations such as temporal tipping points or to specify the regularity of a performance.

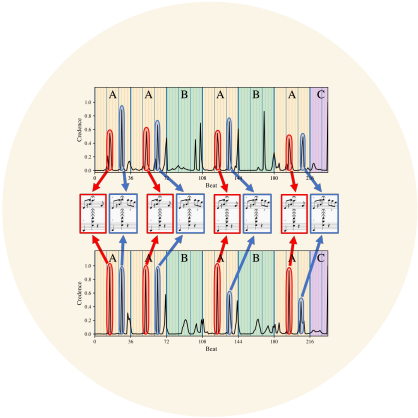
However, this method has some limitations. First, the duration or loudness of each bar is normalized. As a result, information on the total duration or loudness of each bar is not available, i.e. only the proportional distribution of durations and loudness are shown. This means that two points with the same coordinates in the simplex might be two bars of completely different overall loudness or duration. For example, if the three beats of a bar are played equally fast or slow, the corresponding point is at the center of the rhythm simplex. Thus, this method is more appropriate for analyzing performance expressivity at the bar level. Nevertheless, we believe that variations on the scale of a musical phrase should also be considered. The

bars corresponding to the beginning and end of the phrase are points distant from the center of the simplex, as they have increasing or decreasing tempo or loudness values. Whereas the bars in the middle of the phrase are bars with small variations in loudness or tempo, i.e. points in the center of the simplex. Another limitation of this method is that it can only be applied to three-interval data. While this approach is appropriate for the MazurkaBL dataset, which comprises of musical pieces in triple time, it may not be suitable for more general types of data. However, since the MazurkaBL is the largest existing annotated dataset of performed music, we believe that developing tools like the 2-simplex, even if these methods are not applicable to all types of data, is crucial for analyzing and gaining a better understanding of musical performances.

For future research, it would be valuable to study the movement of points within the simplex over time, in order to identify the trajectories of a section of music within the simplex and to capture the variation in interpretations amongst several performers. This would be useful to generate or modify performances by moving points in the 2-simplex and to identify the musical structure of a piece by knowing the trajectory patterns corresponding to the interpretation of sections. Future research using the rhythm simplex in other fields is also possible, for example, to demonstrate that the rhythm of the spoken word is strongly correlated with its content and to discover irregularities in electrocardiograph data. Finally, we expect that the development of methods based on formal mathematical representations, such as the 2-simplex in this chapter, holds great promise for facilitating a more comprehensive analysis and understanding of rhythm in different fields such as spoken voice, electrocardiograph data and musical performance.

Chapter 7

Discovering and Comparing Segmentations in Music Performance Using Tempo and Loudness Information



Contents

In this chapter, we propose a method for discovering and comparing segmentations induced by musical performances. We use a Bayesian model based only on musical prosody data, i.e. tempo or loudness information, to obtain the segmentation of a performance, which is original because we are not using information from the score. In order to discover the similarities between segmentations derived from interpretations of the same piece, we develop a method based on unbalanced optimal transport, and demonstrate on several examples that this approach can be used to analyze and understand particular aspects of musical performance.

This chapter is the result of a joint work with Corentin Guichaoua, a post-doctoral researcher at Ircam during my PhD, who developed the Bayesian model, which is detailed in Appendix B and not in this chapter.

Section 7.1 introduces this chapter by developing the motivations for discovering the segmentations of a musical performance based on musical prosody. Section 7.2 presents the Bayesian model we use to obtain the boundary credence of a performance, shows the different representations our model can generate, and shares some insights from direct examination of the results. Section 7.3 proposes the use of unbalanced optimal transport to reveal similarities and differences between segmentations from different interpretations of the same piece. Finally, we provide some concluding remarks, and point for future developments of this method.

This chapter is an extended version of the article "End-To-End Bayesian Segmentation and Similarity Assessment of Performed Music Tempo and Dynamics With No Score Information" [**?**].

7.1 Introduction

In this section, we explain that a musical performance induces segmentations through musical prosody, i.e. variations of tempo and loudness. We explain that it is possible to discover the induced segmentations of a performance from the musical prosody data alone.

7.1.1 Introduction to Segmentation in Music Performance

When performing a piece, musicians not only have in mind the notes they are about to play, but also some intuitions as to how the musical material, such as notes, group together into coherent ideas, how they relate to each other, and which ideas could be made more prominent during performance. Some of these intuitions may derive from experience, some may be formulated in real-time during performance, parts of it may coalesce into some mental conceptions of the music. All these notions serve to guide the performer's expressive choices which in turn influence how the listener hears the music [Rink 1995]. Using the tools at their disposal, within the constraints of the physical properties of their instrument, the performance conventions they wish

to adopt or reject, and their own technical abilities, performers manipulate timing, articulation and dynamics to shape the music to convey segmentation, prominence, and affect the audience [Palmer 2006]. In particular, a well-documented practice is the *arching* of tempo and loudness to mark phrases, i.e., performers tend to convey phrases through patterns of *accelerando* or *deccelerando* and *crescendo* or *decrescendo* [Gabrielsson 1987, Todd 1992]. Such acoustic variations are referred to as *musical prosody*, which also serve as cues for how the performer or listener segments the musical material. The phrases that performers highlight in this way are approximately non-overlapping and covering the whole piece, hence defining a segmentation. In this chapter, we focus on the problem of discovering and comparing such segmentations from the prosody of recorded musical performances.

7.1.2 Previous Works Related to Segmentation in Music Performance

Segmentation is an important part of perception [Zacks 2007], and it is no surprise that it is widely studied in multimedia research, including for images [Haralick 1985], video [Koprinska 2001], or audio [Sakran 2017]. In particular, music segmentation has received a lot of attention [Paulus 2010, Nieto 2020], with most automatic approaches partitioning the music according to criteria of repetition (Chapter 5) and homogeneity (Chapter 4), as described in Part II. Relatively few methods have focused on musical prosody only as data for discovering segmentation. Widmer and Tobudic fit quadratic models to performance features (instantaneous tempo and loudness)[1] given a known multilevel segmentation [Widmer 2003]. While their aim was not to segment the music, this work highlighted the correspondence between phrase arcs and segmentation boundaries. Chuan and Chew turned the approach around by introducing joint estimation of segmentation boundaries and parameters for an arc model [Chuan 2007], which yields a segmentation solution rather than requiring one. This was later refined by Stowell and Chew who added a Bayesian prior to steer the estimation towards more plausible solutions [Stowell 2013]. Based on this previous work, we choose to focus exclusively on loudness and instantaneous tempo data, discarding all direct score information. This operates a deeper shift than what is immediately obvious: by focusing on musical prosody alone, what is being segmented is no longer the piece as written in the score, but the performance as realized by the musician. Although, as we can observe in our results, the score structure can be partially transferred to the performance, but the extracted structure is of a different nature, barring direct comparisons. Another characteristic which sets apart this work from the existing literature, including that on performance segmentation, is that unlike the above methodologies, we use an end-to-end Bayesian approach, aiming for a credence-based, multiple solution output rather than a single solution.

[1] Instantaneous tempo being the inverse of the time between two beats, and loudness a perceptually adjusted measure of sound pressure [Fastl 2005].

We explain here the motivation for choosing multiple segmentations over the "best" one. Research shows that listeners, when asked to judge the segmentation of a piece of music, sometimes disagree about the exact placement of the boundaries and their existence or relevance [Smith 2014, Nieto 2020]. This indicates that the segmentations that performers project may not be perceived universally the same way. Since part of the disagreement can be traced to listeners focusing on different aspects of the music like rhythm, melody, harmony, or timbre, it seems illusory to expect to recover a sole best projected segmentation based on only one or two features [Smith 2014, Smith 2017]. This calls for a representation of segmentation results that allows for multiple plausible solutions. In short, prior work on music segmentation typically attempts to output a final best guess of the segmentation; even those adopting a Bayesian approach ultimately only output a best answer. By contrast, we aim to provide a more nuanced representation of the segmentation solution in which multiple segmentation hypotheses can co-exist. Such an approach has proven useful in cases where insufficient data are available, as for computer vision [Rupprecht 2017]. These segmentation hypotheses can then be refined, either based on a manual complementary analysis or by using additional sources of data.

To test the algorithm and demonstrate its use on real data, we use the Mazurka-aBL dataset [Kosta 2018]. This dataset, already described and used in the previous chapter, contains about 2000 performances across Chopin's 46 Mazurkas and the corresponding loudness and instantaneous tempo data.

7.2 Boundary Credence Estimation

In this section, we present the model for boundary credence estimation, its outputs, and its interpretation. We show how the complexity of nuanced probabilistic results can be exploited with a few transformations and appropriate visualizations. We then present a few examples to demonstrate how these results can be used to provide a deeper understanding of musical performance.

7.2.1 Model for Boundary Credence Estimation

We have assumed that a performance is driven in part by the performer's segmentation of the piece. However, this segmentation is not directly accessible, as it resides in the performer's mind: it can only be inferred from the data that it has influenced, in particular the tempo and loudness of the performed music, which are easily quantifiable. Therefore, we use Bayesian inference to obtain a model of the plausible segmentations using observed data. To do this, we also need a model of how the segmentation perceived by the interpret is going to drive the data. Thus, the model that we use comprises of two parts: an overall theory of the behavior of segments (i.e. their sizes), and a model of how segmentation decisions affect the prosody (i.e. the arc model of tempo and loudness).

First, we need an overall model of likely segmentations before observing any

data. Sargent et al. used a similar method [Sargent 2017] in popular music. Using their overall model, a segmentation that would divide a piece into a few very short segments and a very long one seems unlikely to be correct. Whereas a segmentation with similar segment sizes or phrase lengths is much more plausible, before even considering the data.

Secondly, a specific model describing how a given segmentation affects the performance data is also required. In the case of loudness or tempo, which exhibit arched shapes delimiting phrases in the context of romantic era music [Gabrielsson 1987, Todd 1992], a piece-wise concave quadratic model appears to be a good compromise between model complexity and modeling error. This specific model drives the data and the kinds of arcs that we are likely to see.

In this chapter, we shall assume that arcs are independent from each other, i.e. that modulations in one arc do not affect those in others, and that the plausibility of an arc depends only on its beginning and end. The two-level model proposed here is similar to the one used by Stowell and Chew, with minor changes to the priors [Stowell 2013]. The main difference is the aim of the computation. Our objective is to discover the posterior credence of all segmentations, summarized through credence values on boundaries, rather than the segmentation of maximal credence.

We detail the model in Appendix B. In this appendix, we first introduce the notation that we used; next, we demonstrate that the output can be efficiently computed from the segmentation prior and the segment-wise data likelihood; then, we describe the arc model we use to compute these likelihoods; we conclude by providing the priors applied in this model.

7.2.2 Visual Representations of the Output

We present here different representations, illustrated in Figure 7.1, which are all derived from the results of our method. In particular, these representations visualize the segment credence outputs of the algorithm. To understand these visualizations, we define $p([i, j] \in S \mid \mathcal{D})$ as the probabilities of having the segment $[i, j]$ in the induced performance segmentation given $\mathcal{D}$ which represent data from musical prosody, which is detailed in Appendix B.

(a) Segment credence matrix: A naive representation of the raw credence values for all possible segments is the *segment credence matrix*, where the (i, j) coefficient is defined by:

$$A_{i,j} = p([i, j] \in S \mid \mathcal{D})$$

Which can be interpreted as: "the probability that a segment starts at i and ends at j". However, this is simple but inefficient, as all non-zero values will be located close to the diagonal, with most of the matrix representing impossible segments. See the bottom left of Figure 7.1 for an example.

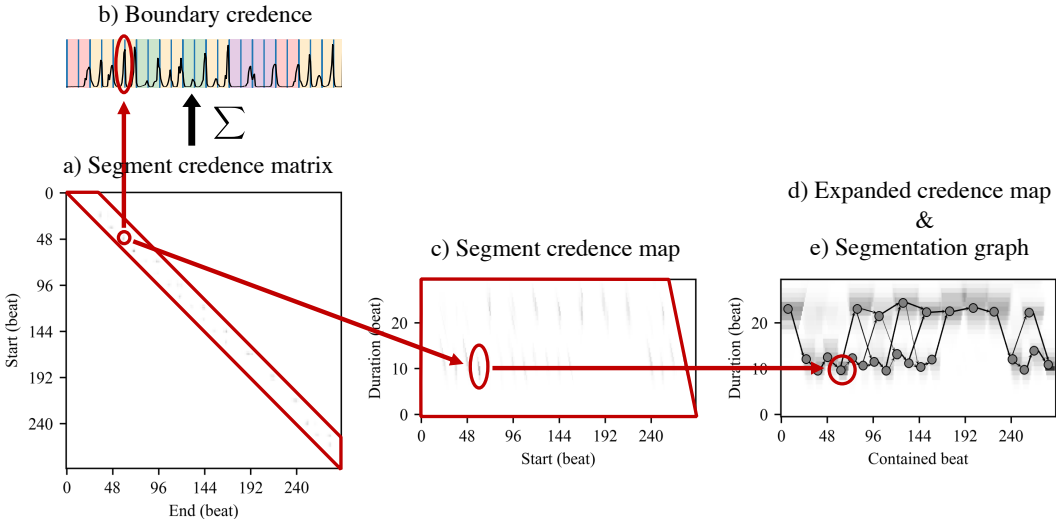


Figure 7.1: Different representations of segment credence for Csalog's interpretation of Mazurka 6-2. a) Raw segment credence matrix. b) Boundary credence. c) Segment credence map. d) Expanded segment credence map. e) Segmentation graph which is a manual conversion superimposed on the expanded credence map, with bolder transitions being more credible. A candidate segment is highlighted throughout the different representations.

(b) Boundary credence: This is a temporal curve that illustrates the credence of each boundary. This yields one real value per beat, similar to the input data, and therefore can be plotted sequentially on the same graph, as in Figure 7.2. For each beat, the boundary credence is defined by:

$$y_i = \sum_{j \geq i} A_{i,j}$$

Which can be interpreted as: "the probability of having a boundary at i". This is probably the most intuitive visualization we provide, and therefore the representation we use in the following.

(c) Segment credence map: The *segment credence map* is a first step towards an efficient representation by transforming the indexing from start and end positions to start position and length of a segment, i.e.:

$$B_{i,k} = p([i, i+k] \in S \mid \mathcal{D})$$

Which can be interpreted as: "the probability that a segment starts at i and has a duration of k". An advantage of this transformation is that the values need only to be computed for acceptable segments. However, it may still be hard to see where each hypothetical segment ends. In addition, many values will still be practically null. See the middle representation in Figure 7.1.

(d) Expanded segment credence map: We can take advantage of the sparsity of the segment credence map by "spreading" each data point over its represented length to obtain the *expanded segment credence map*:

$$C_{k,j} = \sum_{k'=0}^{j} B_{k-j+k',j}$$

Which can be interpreted as: "the probability that k belongs to a segment of length j". An example of this expanded segment credence map is the background representation on the right in Figure 7.1. One can think of each non-zero region as a possible segment, the greater the region's vertical span, the more uncertain its exact boundaries. Vertical region edges indicate sharp boundaries; sloped region edges mark uncertain boundaries.

(e) Segmentation graph: The expanded segment credence map can be manually abstracted as a *segmentation graph*, as shown superimposed on the representation on the right in Figure 7.1. The edges of the graph links pairs of segments that end and begin on successive beats. It provides an overview of the alternative segmentations, but discards information about the precise location of boundaries. Although the segmentation graph was manually created, its construction could be automated.

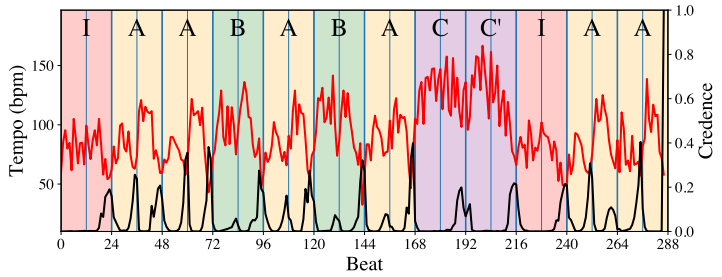
7.2.3 Musical Meaning of Outputs

Here, we discuss two sets of results of the segmentation extraction method. The first shows differences in interpretations of the same musical piece, and the second highlights the detection of expressive gestures in recorded performances.

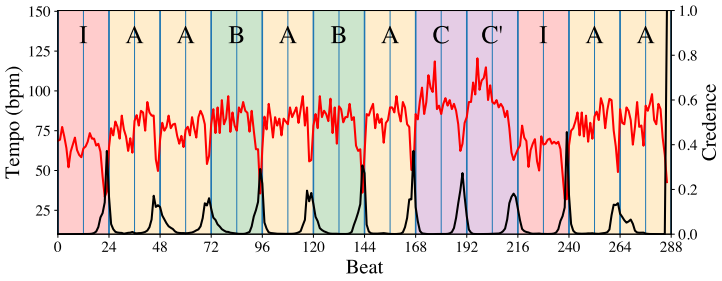
7.2.3.1 Differences in Interpretations of the Same Musical Piece

Figure 7.2 shows the instantaneous tempo values and their derived boundary credence for Gábor Csalog's and Arthur Schoonderwoerd's interpretations of Chopin's Mazurka 6-2. The immediate observation is that, in both cases, boundaries are recovered where section changes occur; the remaining boundaries occur in the middle of sections, where a reference segmentation at a finer scale could have put boundaries. It is not surprising that the performance segmentations align well with a score-based segmentation, as the score structure plays a large role in determining which patterns or groupings can be emphasised. There are nonetheless many differences between the segmentations of each performer.

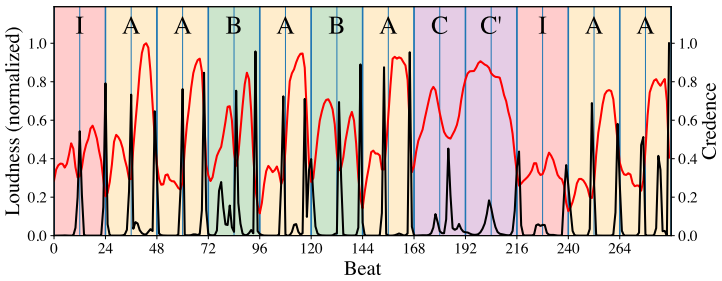
In this instance, based on tempo, the model suggests that Csalog's performance mostly emphasizes 4-bar groupings as seen in Figure 7.2(a), in contrast for example to Schoonderwoerd's performance, where there are 8-bar groupings as shown in Figure 7.2(b). In the broader scheme, Schoonderwoerd also uses loudness to demarcate 4-bars subsections, as shown in Figure 7.2(d), which is picked up by the algorithm when run on loudness.



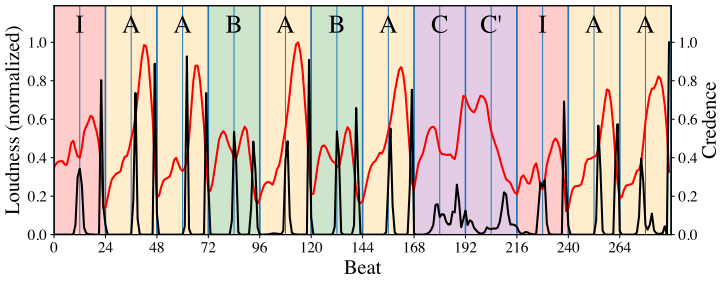
(a) Csalog's interpretation based on tempo



(b) Schoonderwoerd's interpretation based on tempo



(c) Csalog's interpretation based on loudness



(d) Schoonderwoerd's interpretation based on loudness

Figure 7.2: Computed boundary credence applied to interpretations of Mazurka 6-2 of Chopin (in black, the bottom curve). The tempo or loudness raw data are also represented (in red, the top curve). Csalog makes more tempo boundaries than Schoonderwoerd. Csalog's extra boundaries tend to be in the middle of sections; Schoonderwoerd also marks these boundaries but through loudness.

The output also shows that some boundaries are more precisely located than others. For instance, the position of the boundary around beat 168 is sharply defined to within a beat; whereas the next boundary, while still strongly detected overall, is spread out with a loosely defined location. This difference in boundary sharpness can also be observed between the tempo-based and loudness-based segmentations of Schoonderwoerd's performance, mainly due to the smoother nature of the loudness data. With smoother data, the Gaussian noise accounts for less of the variation, leading to tighter fits and thus more confident arc boundaries.

Another interesting feature of Csalog's performance is the weaker boundaries around beats 84, 132 and 156, none of which sum close to 1, reflecting some ambiguity in the structure. Indeed, Csalog weakly marks the 4-bar groupings at these points, but the much higher prominence of the 8-bar arcs could justify skipping the lower-scale boundaries. This is very visible in the segmentation graph on the right in Figure 7.1, where the predominant path uses short segments for the A sections (except the one from beats 144 to 168) and long segments for the other sections, while still showing the alternative long and short sections respectively.

7.2.3.2 Music Expressivity Visualized with Boundary Credence

Figure 7.3 displays the tempo-based output for two performances of Mazurka 24-3 which are Rubinstein's 1966 recording and Fiorentino's recording. Interestingly, the resulting segmentations diverge from expert annotations while largely agreeing with each other[2]. The explanation lies in the presence of *tipping points* in the A sections, i.e. elongations of time for expressive effect [Herremans 2016]. Where boundaries are detected correspond to mid-phrase *fermata* in the score. Indeed, there are tempo arcs starting and ending on these notes, but they arguably do not constitute phrase boundaries, and are in effect temporal tipping points. This result shows that the discovery of the interpreter's segmentation only works so long as the mapping between tempo (or loudness) arcs and musical groupings is not disrupted by other expressive effects. In this case, swapping out the arc model for another which would map correctly to the groupings could be envisioned. Nevertheless, it is interesting to recover expressive gestures like tipping points, which are a form of musical thresholds.

Manual comparison of posterior boundary and segment credences, as we have been doing in this section, is useful and can be enlightening, but it is unscalable to large databases like MazurkaBL and its 2000 recordings for which tens of thousands of pairwise comparisons could be drawn, before even considering cross-piece and cross-feature comparisons. In the next section, we show how we can automatically grade the similarity between performed structures and identify where and how they differ.

[2]This is also the case with other performances of this Mazurka.

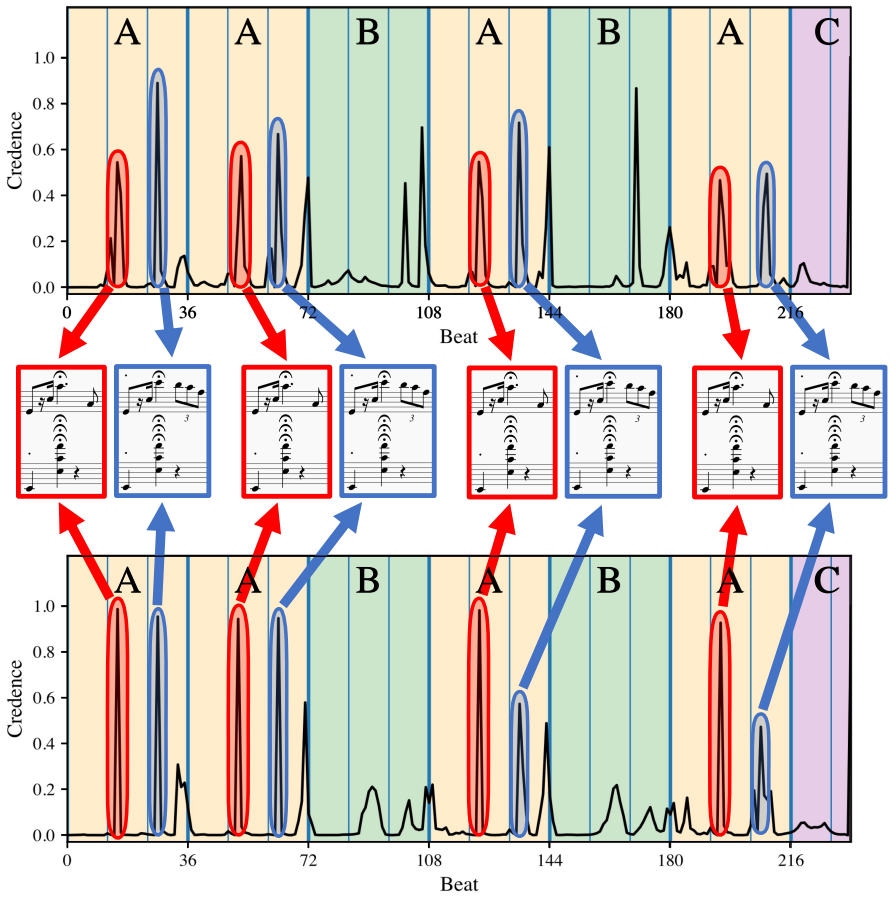


Figure 7.3: Temporal tipping points (suspensions of time flow for expressive effect) found in most performances of Mazurka 24-3 detected as peaks. Examples by Rubinstein 1966 (top) and Fiorentino (bottom). Reference annotation by [Witkowska-Zaremba 2000] (section C is a *codetta* rather than a proper section).

7.3 Distance Based on Unbalanced Optimal Transport to Compare Boundary Credences

In this section, we propose a model to obtain similarities and differences between boundary credences. More precisely, we use unbalanced optimal transport to compute the proximity between the projected segmentations of two performances of the same piece. We then illustrate this method, first by applying it to one of the earlier examples, then systematically to large subsets of the MazurkaBL dataset.

7.3.1 Unbalanced Optimal Transport-Based Distance Model

7.3.1.1 Motivation

With respect to the boundary credence of a given performance, a musician creating another performance may choose to make boundary credence peaks at different locations and with different shapes. For example, compared to Csalog's interpretation in Figure 7.2, Schoonderwoerd chose to create about half as many boundary credence peaks through tempo modulations, marking 8 bar long phrases instead of 4 for Csalog. In addition, the peaks in Schoonderwoerd's interpretation have different shape from those in Csalog's interpretation. Therefore, we propose to quantify the distance between two credence profiles by taking into account the possibility of having a different number of peaks and different shapes for each peak. The two different costs to be accounted for in the distance are:

- **Cost of deforming one peak into another:** When peaks from boundary credence of two different interpretations are found at almost the same locations, they may be of different shapes, eliciting different perceptions from listeners. For example, the perception of a peak can change with a longer or shorter pause between two musical phrases.
- **Cost of destroying or creating peaks:** When a peak is not matched in the comparative performance, this indicates different ways of grouping the music material. The presence or absence of peaks changes the locations and lengths of the musical phrases projected by the performer.

Consequently, each peak is deformed or destroyed and we choose to compute the distance between two boundary credences as the sum of the deformations of the matched peaks into one another and the unmatched peaks that are destroyed or created.

These deformations are computed based on unbalanced optimal transport which is a mathematical theory related to optimal transport [Monge 1781, Kantorovich 1942, Villani 2009]. Optimal transport studies how to transform points from a starting set to an ending set, while minimizing the total cost of transport, where the quantities of the starting and ending set are the same. Unbalanced optimal transport, refers to a situation in which the quantities to be transported and the costs of transport are not balanced among the different sources and destinations. In our case, we are interested in moving the area under a boundary credence to the area under another boundary credence with minimal effort. However, we add the condition that the area under each peak can be transformed to at most one peak. This condition allows us to determine which peaks are similar or different between two boundary credences, which indicate the choices in the way the music is segmented through performance. This method based on unbalanced optimal transport is illustrated in Figure 7.4, and we mathematically formalize this distance in the next subsection.

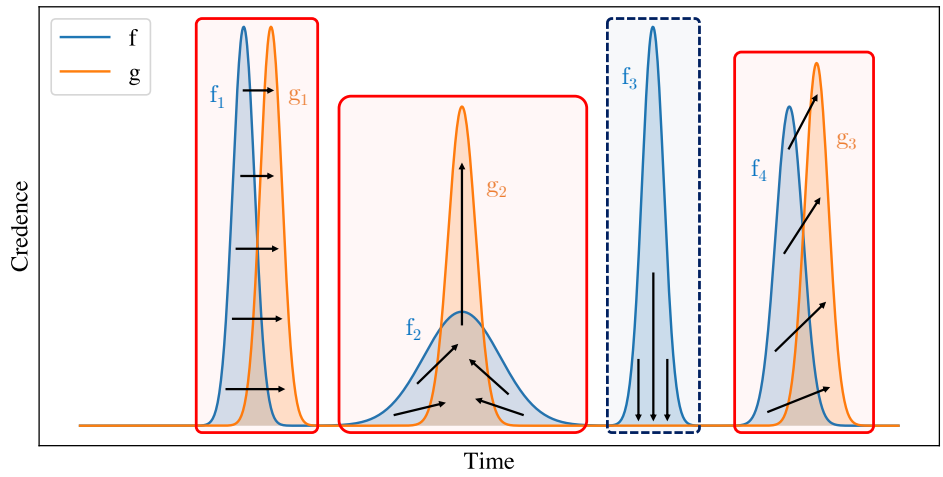


Figure 7.4: Illustration of the unbalanced optimal transport-based distance between two boundary credences f and g. Red rectangles indicate matched peaks; dotted blue rectangles mark unmatched peaks.

7.3.1.2 Mathematical Formulation

Let f and g be two boundary credences as represented in Figure 7.4. Each boundary credence normally comprises of a series of peaks, i.e. $f = \{f_i\}_{i \in I}$ and $g = \{g_j\}_{j \in J}$, where I and J are indices of a set. We isolate the peaks within each boundary credence using a threshold set at 0.01 – values above 0.01 are part of some peak, and those below are not. We have found this value to be sufficient to isolate distinct peaks while discarding only negligible parts of the result. Regions above the threshold give the isolated peaks as illustrated in Figure 7.4 with $f = \{f_1, f_2, f_3, f_4\}$ and $g = \{g_1, g_2, g_3\}$. Given f_i a peak (which is a discrete function), recall that the distribution function F_i of f_i, and the area under the peak $\|f_i\|_1$ of f_i are defined by:

$$F_i(n) = \sum_{k=-\infty}^{n} f_i(k), \qquad \|f_i\|_1 = \sum_{k=-\infty}^{\infty} |f_i(k)|. \tag{7.1}$$

Let f_i and g_j be two peaks such that $\|f_i\|_1 = \|g_j\|_1$. The distance associated with discrete optimal transport[3] d_{OT} [Werman 1985] between f_i and g_j, in the one-dimensional case, is simply written as:

$$d_{OT}(f_i, g_j) = \sum_{n=-\infty}^{\infty} |F_i(n) - G_j(n)|. \tag{7.2}$$

From an algorithmic point of view, distribution functions can be readily computed, so it is straightforward to obtain the optimal transport distance between two peaks.

[3] Also known as Wasserstein's distance, or the Earth mover's distance.

The two peaks may be temporally very distant, in which case it is preferable to destroy the area rather than move it. To do this, we use the unbalanced optimal transport [Chizat 2018]. Let f_i and g_j be two peaks such that $\|f_i\|_1 = \|g_j\|_1$, we propose to define the distance associated with unbalanced optimal transport in the case where the peaks have the same area by:

$$d_{UOT}(f_i, g_j) = \min(\frac{2}{3} d_{OT}(f_i, g_j), \|f_i\|_1 + \|g_j\|_1)$$

According to this definition, if the minimum is $\frac{2}{3} d_{OT}(f_i, g_j)$, it means that it is less costly to transform the area of one peak into another, but if the minimum is $\|f_i\|_1 + \|g_j\|_1$, it means that it is less costly to destroy the area of both peaks. We now explain the 2/3 factor. Because creating or removing a peak translates to a larger musical change than modifying the shape of a peak, we choose to reduce the modification cost between two peaks by a factor of $\frac{2}{3}$. This factor means that it is as costly to match two identical peaks three beats apart with optimal transport as it is to destroy each of the peaks. In other words, the limit for deforming peaks is three beats, i.e. one bar in case of Mazurkas. Mathematically, if $f_i = \delta_{i'}$ and $g_j = \delta_{j'}$ where δ is the discrete dirac function (i.e. $\delta_{i'}(k) = 1$ if $k = i'$ and 0 elsewhere), then: $d_{UOT}(f_i, g_j) = \min(\frac{2}{3}|i' - j'|, 2)$. Therefore, if $|i' - j'| > 3$, i.e. the two peaks are separated by more than 3 beats, then $d_{UOT}(f_i, g_j) = 2 = \|f_i\|_1 + \|g_j\|_1$, so it costs less to destroy both peaks than to deform one into the other. On the other hand, if $|i' - j'| < 3$, i.e. both peaks are less than 3 beats away, then $d_{UOT}(f_i, g_j) = \frac{2}{3}|i' - j'| = \frac{2}{3} d_{OT}(f_i, g_j)$ and the cost of transforming one peak into the other is lower than destroying the peaks.

Finally, when two peaks, f_i and g_j, do not have the same area, for example with $\|f_i\|_1 \leq \|g_j\|_1$, we scale the area of g_j with the factor $\frac{\|f_i\|_1}{\|g_j\|_1}$ so that the two peaks have the same area and we add a term to signify the cost of the area lost, $\|g_j\|_1 - \|f_i\|_1$, as already proposed by Gromov for the optimal transport formula when areas are different [Gromov 1999] (chapter 3 1/2, section B, page 117). We then propose to define a distance based on unbalanced optimal transport when two peaks have different areas with Definition 7.1.

Definition 7.1: Unbalanced optimal transport-based distance

Let f_i and g_j be two peaks such that $\|f_i\|_1 \leq \|g_j\|_1$, we define the **unbalanced optimal transport-based distance** between f_i and g_j by:

$$d_{UOT}(f_i, g_j) = \min(\tfrac{2}{3} d_{OT}(f_i, \tfrac{\|f_i\|_1}{\|g_j\|_1} g_j), 2\|f_i\|_1) + \|g_j\|_1 - \|f_i\|_1$$

Referring back to Figure 7.4, some f peaks are matched and moved to g peaks (red rectangles) and others are destroyed (dotted blue rectangle). Recall that each

peak can only be matched once. We use a dynamic time warping algorithm to determine which peaks of f should be matched, or not, with those of g by allowing peaks to be matched with the zero-value function, which is equivalent to peak destruction. In Figure 7.4, the first peak f_1 of f is slightly shifted toward the first peak g_1 of g, the second peak f_2 is deformed because it does not have the same shape as g_2, and the third peak f_3 is destroyed because it does not match with any peak of g. In the end, the distance between two boundary credences is the sum of the cost of transforming or destroying each peak.

7.3.2 Revealing Similarities and Differences in Segmentations of Different Interpretations

We illustrate the distance based on unbalanced optimal transport using the Csalog's and Schoonderwoerd's interpretations of the Mazurka 6-2 shown in Figure 7.2. The comparison result is shown in Figure 7.5.

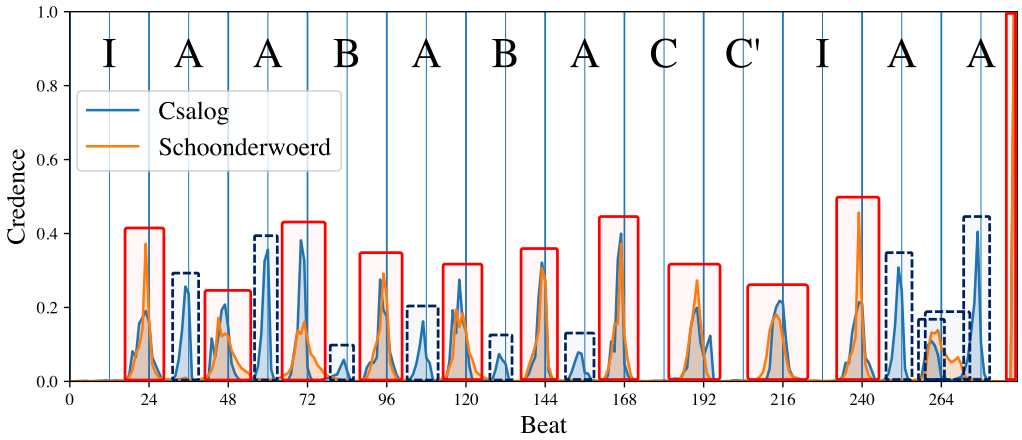


Figure 7.5: Unbalanced optimal transport-based distance between the interpretations of the Mazurka 6-2 by Csalog and Schoonderwoerd. Note that the red rectangles indicating agreement tend to align with the annotated section boundaries, while dotted blue rectangles mark more unusual boundaries.

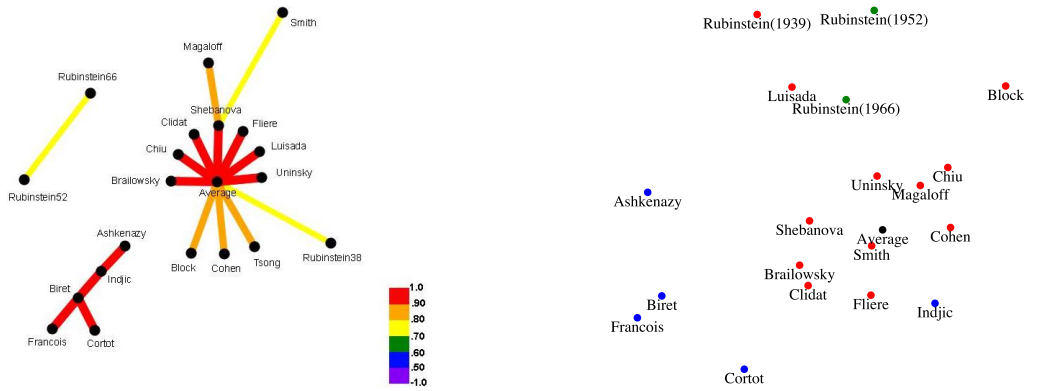
In Figure 7.5, we can understand which peaks are matched (marked by red rectangles) between the two performances and which peaks are removed (delineated with dotted blue rectangles). This can be useful for understanding the similarities and differences between two recorded performances. Because there are peaks between two successives sections in both performances (i.e. every 8 bars), they are deformed into each other through unbalanced optimal transport. By contrary, most of the additional peaks of Csalog compared to Schoonderwoerd are destroyed. It is also interesting to notice that even if the second to last peaks of each performances are almost at the same time, they do not match. The algorithm prefers destroying them rather than deforming them because their shapes are too different. Also,

with respect to the last peak, the two curves are overlapped, so they cannot be distinguished in the figure and the peaks are matched with the unbalanced optimal transport-based distance (indicated by a red rectangle).

7.3.3 Detecting Similarities on a Large Scale

Cook investigated the tempo variations of different performances of the Mazurka 68-3 [Cook 2007]. He computed correlations between the raw tempo curves of the different performances and manually represented those that are highly correlated by a *correlation network*, illustrated in Figure 7.6(a). He identified three clusters, thus three main different ways of playing Mazurka 68-3 that he explained partially by geographical location or teacher/pupil relationships between the different pianists. We then automated the computation of this correlation network of the same performances (except for Tsong who is not in our database).

We first computed the credence boundaries and then the distance between them based on the unbalanced optimal transport as described in the previous section. Finally, we automatically represent the result with a *similarity map* in Figure 7.6(b) using the Python package *manifold* from the *scikit-learn* module [Pedregosa 2011]. We assigned a color to each of the three clusters established by Cook. The result presented in Figure 7.6(b) is consistent with the analysis of [Cook 2007]. For example, we can observe that Rubinstein's interpretations are far from the average. In addition, there are other interpretations that are far from the average, as Cook had mentioned, namely Ashkenazy, Biret, François and Cortot.



(a) Manually created correlation network by Cook [Cook 2007]

(b) Automatically generated similarity map based on our method

Figure 7.6: Relation between different interpretations of Mazurka 68-3.

We then tried to obtain further results on the relations between interpretations of other Mazurkas. For this, we focused on Arthur Rubinstein, because in the MazurkaBL database, he stands out as by far the performer with the most recordings. He recorded three sets of Mazurka performances: in 1939, 1952, and 1966. All three covered most, if not all, of the Mazurkas. Although his style evolved over the years,

his performances remained on average closer to his own than to others', according to our distance measure, as can be observed in Figure 7.6(b). Re-scaling the distance such that the closest performance pair on a piece is 0 and the farthest is 1, the average distance between Rubinstein's recordings of the same piece is 0.28, whereas that between Rubinstein's recording and another performer's is 0.52.

Extending this idea, it would seem reasonable to hypothesize that trends of proximity between artists can persist across pieces, perhaps due to similar grouping preferences or other similarities in their structural perception. To test this hypothesis, we focused on the subset of performers who recorded all of the Mazurkas that Rubinstein recorded on all of his three sets (overall 30 Mazurkas and 10 performers in addition to Rubinstein's three versions). We then looked at the 30 distance matrices for each Mazurka, and proceeded to perform a Mantel test [Mantel 1967] on each pair of Mazurkas. However, only 39 of the 435 pairs showed significant correlation at the 5% level, which is higher than would be expected by chance, but far below what might result from a sizeable trend.

7.4 Conclusion and Future Work

In this chapter, we have described a method aimed at discovering the implicit segmentation conveyed through a musical performance. To achieve this, we have relied on a Bayesian framework, which has led to a nuanced output in which many segmentation hypotheses can co-exist. The method works on extracted prosodic features of an audio recording of a performance, without the need for score information. The nuance implies that with limited features and segmentation ambiguity, it may not be possible nor desirable to have precise localization of boundaries, and also that more than one segmentation can be a valid explanation for the observed data. We have demonstrated on a selection of examples that this method discovers segmentations reflecting differences between performances.

We have also proposed a method based on unbalanced optimal transport which yields a distance between performed structures and highlights their similarities and dissimilarities. Interestingly, this distance revealed that Rubinstein's performed structures across the years were more similar to each other than to other pianists'. By contrast, we have found no significant correlation between two performers' distance in one piece and their distance in another. This means that agreeing on performed structures in one piece may not imply agreement in another piece. However, it is important to recall that the comparisons mentioned have been based only on tempo or loudness segmentations. Indeed, two performances can be similar in these aspects but may be different on other counts like overall tempo or timbre.

In future work, it would be desirable to apply this method to a larger database of performances, preferably annotated with perceived structures. For example, the ASAP dataset [Foscarin 2020] has a broader composer and piece range than the MazurkaBL dataset [Kosta 2018] we used, at the cost of a shallower range in performers. This broader range likely includes some pieces for which conventional

interpretations do not exhibit the arching patterns we rely on, requiring a different segment model. Unfortunately, it still does not include performance structure annotations, and to our knowledge, neither does any currently available database. Also, there is some qualitative evidence that performance structure could be used as an approximation of score structure, which encourages further research.

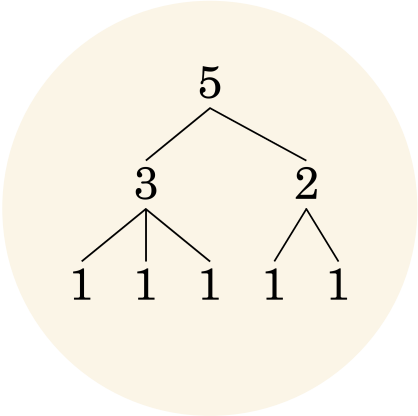
We believe that, beyond the specifics of the model and algorithm presented, one of the key takeaways of this chapter is the choice of credence on boundaries or segments as outputs of segmentation. It has the potential to give deeper insights into music, for example the distinction between certain or optional boundaries, and between strongly and weakly localized boundaries. In order to make full use of this rich output, new visualizations, tools and methodologies are critical. This is what we have aimed to achieve in this chapter with new musical representations and by adapting some mathematical theory to better model and understand musical performance.

Part IV

Science Outreach

Chapter 8

Science Outreach for "Mathémusique"



Contents

In addition to my PhD, I have developed a science outreach activity to explain mathematical concepts applied to music. I mainly realized this activity on YouTube with the *"Mathémusique"* channel available via the link: `https://www.youtube.com/@mathemusique`. This chapter focuses exclusively on the two

major representations used in this science outreach activity: the *rhythmic tree representation*, developed to visualize the structure of a musical theme in a hierarchical way, and the *circular representation*, mainly used to analyze the properties of musical rhythms.

Section 8.1 presents the advantages and novelties of the rhythmic tree representation and how it relates to existing representations of musical structure and meter. Section 8.2 describes an open-access interactive website developed to generate rhythms using a circular representation.

8.1 Temporal Hierarchies of Music Represented as Rhythmic Trees

This section presents the rhythmic tree representation for visualizing the hierarchical structure of a musical theme. Starting with an introduction to hierarchical temporal representations of music and demonstrating the originality of this visualization. Then, describing the advantages of this representation and how it has been used in our science outreach activity.

8.1.1 Introduction to Hierarchical Temporal Representations of Music

In the book *Generative Theory of Tonal Music* (GTTM), Lerdahl and Jackendoff proposed a tree representations, *time span tree* and *prolongation tree*, to illustrate the hierarchical structure of a musical theme [Lerdahl 1985]. The time span tree representation is based on metrical structure (smallest levels) and grouping structure (larger levels), defined through Lerdahl and Jackendoff's generative rules. It is illustrated with Mozart's K. 331 theme in Figure 8.1.

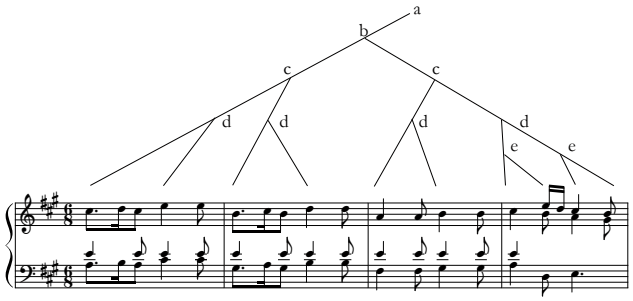
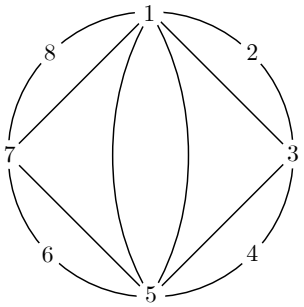


Figure 8.1: Example of Lerdahl and Jackendoff's tree representation of Mozart's K. 331 theme. (Source: [Lerdahl 1985], 172.)

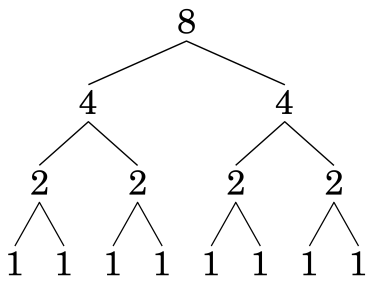
On the other hand, the prolongation tree is based on the structure of tension and relaxation in the piece of music. The representation is similar to the time span tree representation, with the exception of nodes indicating strong prolongation, weak

prolongation or a progression. These representations are a first attempt to visualize the structure of musical themes in a hierarchical way.

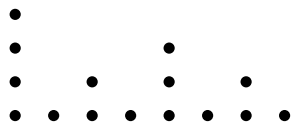
However, we believe that these GTTM tree representations could be more intuitive, due to the large number of branches and their different directions. We have therefore proposed a visualization of the hierarchical structure of a musical theme, inspired by the different existing hierarchical representations of meter illustrated in Figure 8.2. These representations are equivalent, but have different visualization advantages. The *metrical structure* detailed in the GTTM, makes it easy to visualize the strong and weak beats of a musical bar [Lerdahl 1985]. The *cyclical representation* helps us understand the cyclical nature of meter [London 2012]. The *maximal outplanar graph* representation is similar to the cyclical representation redraw with straight line [Yust 2018]. The *nested brackets* are the most condensed representation of this hierarchy [Gotham 2015]. Finally, the *tree structure* [Gotham 2015] or *tree diagram* [Hanenberg 2018] representation is the closest to the GTTM tree representation.



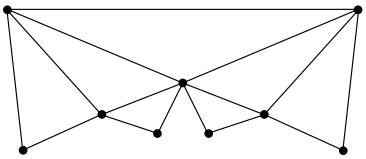
(a) Cyclical representation



(b) Tree structure or tree diagram



(c) Metrical structure



(d) Maximal outplanar graph

((2,2),(2,2))

(e) Nested Brackets

Figure 8.2: Different hierarchical representations of the $\frac{4}{4}$ meter.

Therefore, to represent the hierarchical structure of a musical theme, we adapt the tree structure representation in Figure 8.2(b) to the duration of a musical theme, using the Lerdahl and Jackendoff rules described in the GTTM to produce the time span tree. The resulting representation, called a *rhythmic tree*[1], is shown in Figure 8.3. As with the time span tree representation, the smallest hierarchical levels are based on the metrical structure, while the largest ones are based on the grouping

[1]This representation is also called *"arbre rythmique"* (french translation).

structure. However, unlike the GTTM tree representations, a number indicates the duration for each node in the tree, and each hierarchical level is clearly represented by a line of the rhythmic tree (which is not true for other representations such as cyclical, maximal outplanar graph and nested bracket representations). Moreover, the advantage of the rhythmic tree representation for an outreach activity is that it uses very elementary symbols (numbers and lines) rather than musical symbols which may be more complicated to understand for an unfamiliar public.

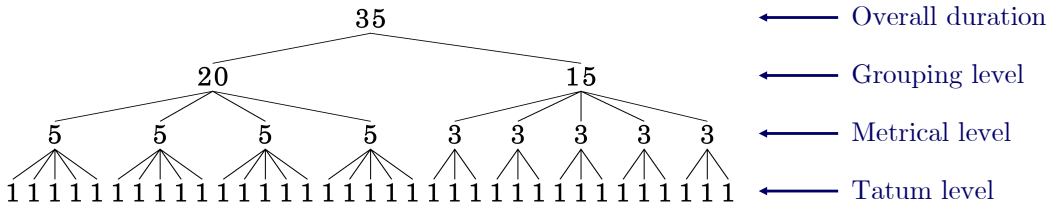


Figure 8.3: Rhythmic tree representation of *Entertain Me* by Tigran Hamasyan with four hierarchical levels.

The difference with the tree structure representation is that we apply it not just to the meter, but to a musical theme. As a result, the visualization has to be adapted, which is not trivial, as Gotham points out that to build metric hierarchical structures, groups of 2 or 3 have to be formed [Gotham 2015]. However, this is not what we have done in Figure 8.3, as the metric evolves in the musical theme, so we have groups of 3 and 5 in the last line of the tree. This is why we use the rules of the GTTM, with the difference that we remove some hierarchical levels from the tree (i.e. we remove some lines) to avoid having too much information. Because with too many hierarchical levels (with more than 4 or 5, judging by ourselves), we lose visual interest and it is no longer intuitive. As a result, the most important hierarchical levels are represented to ensure the best possible understanding when visualizing the rhythmic tree. In the following sections, we present the advantages of this representation for intuitively visualizing certain characteristics of a musical theme.

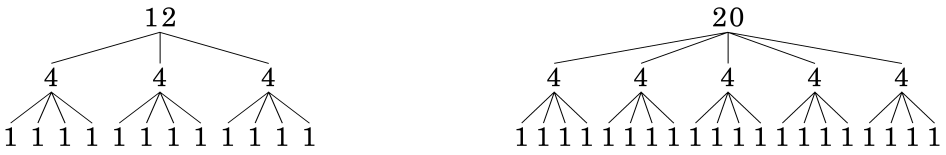
8.1.2 Advantage 1 of Rhythmic Trees: Groupings and Overall Duration on the Same Representation

One of the main advantages of the rhythmic tree representation is that the overall duration and the durations of the note groups are clearly indicated on this representation. The overall duration is naturally displayed at the top of the tree, while the durations of the note groups are indicated by the numbers on the lower lines. For example, in Figure 8.3, the overall duration is indicated by the number 35, and we understand that this theme can be divided in two by a group of duration 20 (which is itself divided into blocks of duration 5) and by a group of duration 15 (which is

itself divided into blocks of 3). It may seem essential to have overall duration and group durations on the same representation, but this is not the case for the previous representations presented in Figure 8.1 and Figure 8.2, with the exception of the tree structure representation, from which we mainly took our inspiration.

8.1.3 Advantage 2 of Rhythmic Trees: Provides Information on a Hierarchical Level Above the Bar

Another advantage of the rhythmic tree representation is that it produces information above the bar, unlike the representations presented in Figure 8.2, which are designed for metric representation only. This is relevant when describing a musical theme, as it is often composed of several bars. In particular, it makes it easy to visualize the number and type of bars contained in the theme. For example, themes composed of 3 bars or 5 bars of 4 beats are easily identifiable, as shown in Figure 8.4. In addition, if the type of bar changes, for example by composing 3- and 4-beat bars, this is displayed very clearly on the rhythmic tree representation.



(a) 3-bar theme

(b) 5-bar theme

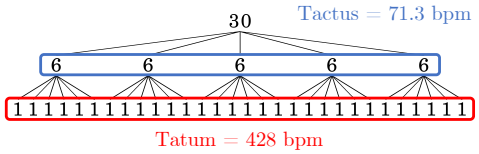
Figure 8.4: The rhythmic tree representation provides information on a hierarchical level above the bar, which is relevant for themes of 3 or 5 bars long.

8.1.4 Advantage 3 of Rhythmic Trees: Musical Transformations of Each Hierarchical Level are Visualized

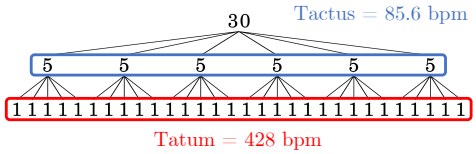
The rhythmic tree representation also makes it possible to visualize certain musical transformations, since changes can be seen on each line of the tree. For example, Figure 8.5 illustrates tactus changes in the song *Lights Out* by Car Bomb, resulting in a modification of the note groups on the middle line. In this song, there are first groups of 6, then 5, 4 and finally 3 tatum pulsations. The overall duration does not change, so only the middle line is modified. Therefore, this transformation is easily illustrated with the rhythmic tree representation.

It is also possible to visualize tatum changes, as shown in Figure 8.6 with the song *Story 2* by Clipping. In this case, the middle line is still made up of 4 groups, but the numbers have changed because the tatum has changed. Another transformation in this song, which is also very easy to visualize with the rhythmic tree representation, is the change of metric from $\frac{3}{8}$, then $\frac{4}{8}$, $\frac{5}{8}$, $\frac{6}{8}$, $\frac{7}{8}$ up to $\frac{8}{8}$. The visualization of this song

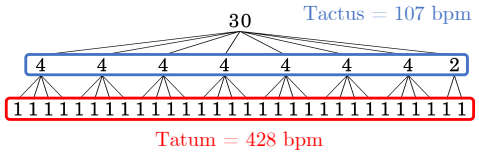
with the rhythmic tree representation is available via the link: https://youtu.be/Hyvv1oJAaSI?si=fs64lW30BHYJDI34.



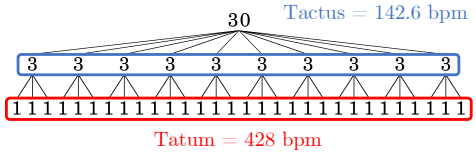
(a) Groups of 6 tatum pulses (at 0:08)



(b) Groups of 5 tatum pulses (at 0:12)

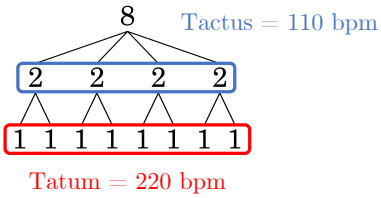


(c) Groups of 4 tatum pulses (at 0:16)

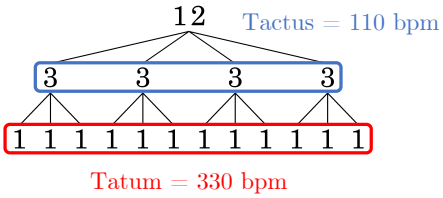


(d) Groups of 3 tatum pulses (at 0:21)

Figure 8.5: Musical transformation in the introduction of *Lights Out* by Car Bomb visualized with the rhythmic tree representation. Only one of the three hierarchical levels is modified (the tactus).



(a) Tatum before 1:15

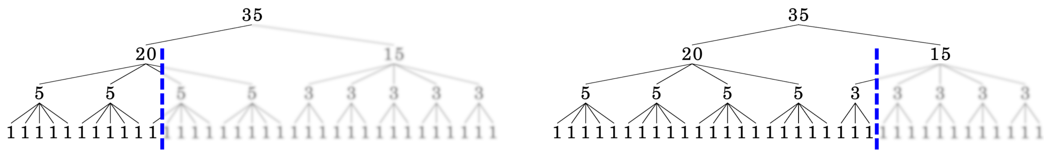


(b) Tatum after 1:15 (3/2 faster)

Figure 8.6: Musical transformation of *Story 2* by Clipping (at 1:15) visualized with the rhythmic tree representation. Only one of the three hierarchical levels is modified (the tatum).

8.1.5 Advantage 4 of Rhythmic Trees: Visualize Truncated Patterns

Since the time axis is present in the rhythmic tree representation, i.e. the tree is interpreted from left to right, it is simple to represent truncated versions of the musical theme. This technique is frequently used by some musicians [Pieslak 2007]. For example, Figure 8.7, illustrates two truncated versions of the theme from the song *Entertain Me* by Tigran Hamasyan. The visualization of this song with the rhythmic tree representation is available via the link: https://youtu.be/EidE2ETpCnU?si=TbtJK7rV3mQFZT42.



(a) First truncated theme (at 0:28) (b) Second truncated theme (at 1:10)

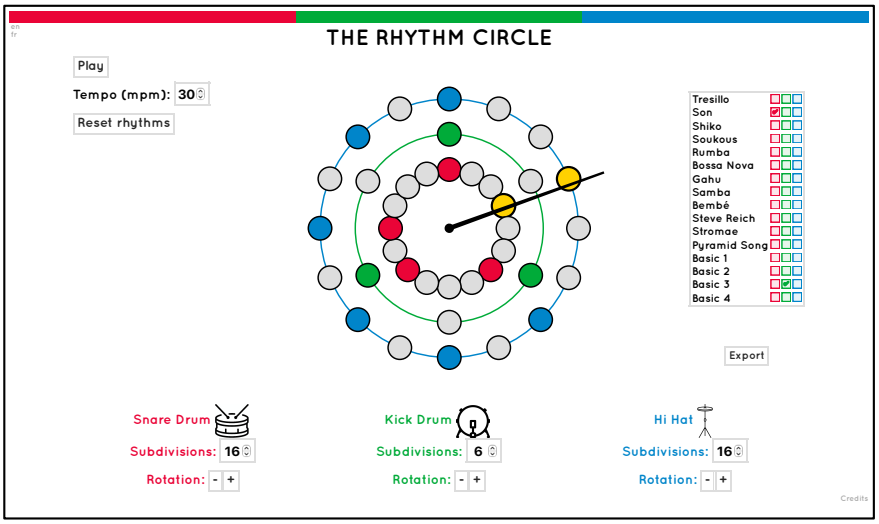
Figure 8.7: Visualization of truncated themes of *Entertain Me* by Tigran Hamasyan using the rhythmic tree representation (the complete theme is represented Figure 8.3).

8.2 Interactive Software Developed to Generate Rhythms Using a Circular Representation

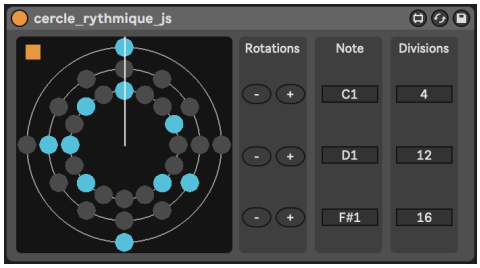
Circular representation is a powerful method of making direct links between musical and geometric objects. Specifically, Toussaint analyzed the properties of musical rhythms by considering the properties of the corresponding polygons when these rhythms are represented within a circle [Toussaint 2005, Toussaint 2019]. Many of these ideas have been incorporated into our science outreach activities.

After introducing the circular representation, I received numerous messages from teachers of both mathematics and music, asking me if there was any user-friendly, free software they could use to create projects based on the circular representation with their students. I therefore decided to start a project to create a free interactive website that would enable the general public to use this representation. I was then selected by the French Musicology Society (SFM) and the French Association of Computer Music (AFIM) for the call for projects of the French Ministry of Culture "Research in Music: Collaborations between Young Researchers and Artists" to develop the project "Articulation between Theoretical Research and Algorithmic Composition". This allowed me, in part, to develop in collaboration the interactive website illustrated by a screenshot in Figure 8.8(a).

This website is available via the link: `https://mathemusique.gitlab.io/rhythm-circle/`. There is also a version of a midi effect for Ableton Live that has been developed for musicians which is also illustrated by a screenshot in Figure 8.8(b). These software incorporate the six distinguished rhythms considered by Toussaint [Toussaint 2019] presented in Section 1.3.2. The most interesting feature of these interactive software is that you can choose any subdivision of the circle. For example, you can combine a subdivision of 12 with 8 as in Figure 8.8, but also other more complex subdivisions such as between 11 and 7. In addition, known rhythms can be immersed in unusual divisions, for example the six distinguished rhythms can be dropped into a subdivision of the circle of 17 to create an unnatural and artificial groove.



(a) Interactive website



(b) Midi effect in Ableton

Figure 8.8: Interactive software called "The Rhythm Circle" developed to generate rhythms using a circular representation.

8.3 Conclusion

I learned a lot from this science outreach activity. This led me to develop my curiosity in many subjects of mathematics and music, and to synthesize information to transmit it to the general public. I would like to point out that I did this in my spare time, mostly at weekends. Given the interest shown by people, I intend to continue doing this in the years to come, as long as I remain in the academic domain. Finally, I would like to list the three videos that have reached over 300,000 views on YouTube, which are: *What Are Euclidean Rhythms?*[2], *Minimalist Music, an Intellectual Art Form?*[3], and *Meshuggah and Algorithmic Music*[4]. These three videos are in my native language, in French, and I recommend watching them for a better understanding of this science outreach activity.

[2] https://youtu.be/8G8qko7NZdE?si=gkudTeJs6mn3Ka63
[3] https://youtu.be/47agCNyxfOE?si=f0kZEmHdHQmk72-Q
[4] https://youtu.be/euSki63c-SQ?si=IsZRciltMrnbiHwc

Conclusion and Perspectives

Conclusion and Perspectives

To conclude, we develop a discussion on all three parts of this manuscript in relation to our contributions and detail the perspectives of this research. Finally, we propose a response to the relevance of applying mathematical theories to music.

Part I. Musical Pattern Discovery

The first part of this manuscript proposes a method to discover musical patterns from a multidimensional representation of music. The originality of this method comes from the use of mathematical morphology, a theory rarely applied to music. Therefore, before presenting our method to discover musical patterns, we first chose to introduce mathematical morphology and previous works on musical pattern discovery from a multidimensional representation of music.

Chapter 1 introduced the theory of mathematical morphology. We have considered only the binary framework, which is the simplest case, and presented the main operators and some of their properties. The contributions of this chapter are the multiple applications to symbolic representations of music that we have suggested. In particular, we have demonstrated that the principal operators can produce musically relevant results on musical chords (with $\mathbb{Z}$), musical rhythms (with $\mathbb{Z}_n$) or musical pieces (with $\mathbb{R}^n$). The remainder of this part is based on the application of mathematical morphology to musical pieces.

Chapter 2 presented previous works on the discovery of musical patterns from a multidimensional representation of music. Our contributions have demonstrated that mathematical morphology fits perfectly into this context. In particular, we have proved that the principal operators of mathematical morphology generalize and provide a deeper understanding of certain fundamental concepts of the musical pattern discovery task. Therefore, these first two chapters of this part set the context for our method, which is presented in the next chapter.

Chapter 3 described our method for discovering musical patterns from a multidimensional representation of music using mathematical morphology. The contributions of this chapter are the various mathematical results that are useful for the musical pattern discovery task. We have shown that these results can optimize the discovery of musical patterns and provide a response to some of the open questions in this field such as the discovery of MTEC patterns or the "problem of isolated membership". In addition, these results are original because they come from musical problems and can be applied to other types of data.

In my opinion, this is the most accomplished part of this manuscript. This is the part I most enjoyed working on and the part that I am most proud of. For future work, Chapter 1 can be extended as we have only considered the principal

operators of mathematical morphology, but we believe that other morphological operators should be considered for possible application to music. In Chapter 2, the majority of existing algorithms for the discovery of musical patterns have already been reviewed, so it seems complicated to produce new links between morphology and these algorithms. Finally, we are convinced that Chapter 3 is the chapter to be developed in this manuscript. Since this chapter proposes a new approach for discovering musical patterns, it provides multiple perspectives for future work, such as those suggested in Section 3.6.2 (with the discovery of non-repeating patterns, truncated patterns, meter, or musical sections). Addressing these questions opens up the possibility of generating additional mathematical results from musical problems. More generally, the results developed in this chapter can also be applied to other types of data, such as a signal-type representation by using a spectrogram and adapting the results, or simply to discover repetitions in a discrete set of points.

Finally, to answer the original question stated in the introduction of this manuscript: *How to discover patterns from a multidimensional representation of music?* The short answer is to use the onsets. The long answer is that it is possible to give a musical meaning to the operations used and the patterns discovered by distinguishing the role of onsets and musical patterns. This reveals that the onsets and musical patterns are MTEC conjugated and therefore form pairs. This implies that it is possible to discover the musical patterns from the onsets. We then proposed to focus on non-redundant pairs, as they can describe musical sections, and several such pairs can provide a good approximation of a musical piece.

Part II. Musical Structure

The second part of this manuscript deals with the generation of hierarchical segmentations from a symbolic representation of music. Based on previous works, the main criteria to consider are: homogeneity, novelty and repetition. We then proposed a method based on homogeneity and novelty, which are closely related criteria, and another based on repetition.

Chapter 4 focused on the homogeneity and novelty criteria for generating segmentations in a hierarchical way. Our contributions have shown the relevance of morphological filters in the grayscale framework for homogenizing data. In particular, we have illustrated that these filters can homogenize regions in self-distance or self-similarity matrices, and that the size of the filter changes the level of homogenization.

Chapter 5 proposed the generation of hierarchical segmentations based on repetition. The contributions of this chapter are first the generalization of an existing representation (the correlative matrix) to discover almost repeating patterns without overlaps in a musical sequence, and then develop a method that extracts data from it to produce hierarchical segmentations.

In my opinion, this part is original because it uses a symbolic representation of music and offers new tools for generating segmentations in a hierarchical way. However, the main weakness of this part is that it is based on a string representation of music. This representation is useful for generating the self-similarity matrix, for example, but is not robust, in the sense that a slight change in the data can lead to a large change in the obtained result. Therefore, in future work, we believe that it is important to work with a multidimensional representation of music, as in Part I, and to consider homogeneity and repetition criteria together in the same method. Additional work remains to be done for the homogeneity criterion, but for the repetition criterion, the results developed in Part I can be used, which would constitute an original approach to discovering the structure of a musical piece.

Finally, to answer the original question stated in the introduction of this manuscript: *How to discover hierarchical segmentations from symbolic representations?* The criteria to be considered are homogeneity, novelty and repetition. We have extended some previous works that use a string representation of music and have validated that these criteria are useful for the discovery of hierarchical segmentations. However, with a symbolic representation of music, we believe that a method based on a multidimensional representation of music is required, and that this method should take into account all these criteria together and not just one of them.

Part III. Musical Performance

The third part of this manuscript analyzes musical performance. Since this is a very complex subject, we have chosen to focus on tempo and loudness variations in solo piano performances from the MazurkaBL dataset. This choice is original because we are not using information from the score. In this part, we propose to analyze these performances in two different ways.

Chapter 6 proposed a new representation of musical performance. Our contributions have demonstrated the relevance of the rhythm or loudness simplex to represent the performances of the MazurkaBL dataset. With this representation, we have shown how to obtain bars with a high expressivity, or how to compute the regularity of a performance.

Chapter 7 analyzed and compared segmentations derived from musical performances. As this work comes from a collaboration, I have mainly detailed my contribution, which proposes to use unbalanced optimal transport to compare two musical interpretations of the same piece. This method reveals similarities and differences between two interpretations.

In my opinion, the analytical approaches to musical performance developed in this part are innovative. I am satisfied to have represented the performance in a 2-simplex, and to have found an original musical application for unbalanced optimal

transport. However, the process of musical performance is still far from being fully understood, and much work remains to be done. An interesting extension to this part would be to establish links between the patterns and structure discovered in the score, and the segmentation induced by interpretations of the same piece. We therefore believe that it would be relevant to use the results developed in Part I and Part II and to compare them with those of Part III of this manuscript.

Finally, to answer the original question stated in the introduction of this manuscript: *How to analyze musical performance using tempo and loudness information?* We first proposed to represent the data in the rhythm simplex (for tempo data) or loudness simplex (for loudness data). We have shown that the isolated points in this representation correspond to bars with high expressivity. We then developed a method for recovering the segmentation induced by a performance by considering variations in tempo and loudness data. This can be used, for example, to display musical phrases or temporal suspensions from an interpretation. Finally, we have shown in this part that many of the expressive choices made in a musical performance are reflected in tempo or loudness data.

Final Remark

In this manuscript, we have applied many different mathematical theories to music. In particular, we have shown that it is possible to give a musical meaning to mathematical operations, and reciprocally, that it is possible to characterize musical objects with mathematics. If we were able to do this, we might first think that it is because "music is math", and therefore it is completely obvious to explain all musical concepts using mathematical language. However, this is not true. Whether it is timbre, emotions, production, improvisation, social interaction, or political and cultural context, there are many musical properties that we have not covered in this manuscript because mathematics may not always be the most appropriate approach. What can we conclude from this? We can therefore conclude that the statement "music is math" is a bit simplistic, but this manuscript is not 200 pages long just to conclude that "music is music, and math is math". If there is a deep connection between mathematics and music, it is partly because many musical objects can easily be described by symbols. In particular, there is a whole musical heritage that comes from writing music using a score, where symbols can easily be described in a mathematical framework. This makes it possible, for example, to describe a musical note by a point in $\mathbb{R}^n$ as we have done in Chapter 1, Chapter 2, Chapter 3, Chapter 4, and Chapter 5 of this manuscript. But symbols are not limited to notes: they can also describe structure, rhythm, chords, meter, texture, tonality and many other fundamental musical concepts. With this abstract description, it is then possible to use a mathematical formalism and discover links between mathematics and music.

Appendix

Appendix A

Theorem and Proof of the Existence of a Conjugate Pair With a Non-Periodic Pattern

We prove here Theorem 3.8 recalled below, which assures the existence of a non-redundant MTEC conjugate pair when a non-periodic pattern is repeated at regular intervals.

Remark (Notation). In the following, we represent the first coordinate of $x \in E$ by t_x, which refers to the temporal component and for $S \in \mathcal{P}(E)$, we define $S_{[a,b]} = S \cap \{p \in E \mid t_a \leq t_p \leq t_b\}$.

Theorem 3.8: MTEC conjugate pair with a non-periodic pattern

Let $X = P \oplus O$ with $P, O \in \mathcal{P}(E)$ such that:

- $L = (t_L, 0, \dots, 0) \in E$ with $t_L > 0$,
- P is of temporal length less than t_L (i.e. $\max\limits_{x,y \in P} |t_x - t_y| < t_L$),
- $O = \{kL = (kt_L, 0, \dots, 0) \mid k \in [\![0, N-1]\!]\}$ with $N \in \mathbb{N}^*$,
- P is non-periodic in all directions (meaning $\nexists S \in \mathcal{P}(E)$ such that $S \subsetneq P$, S is not a singleton and $\gamma_S(P) = P$).

With these assumptions, we have:

(P, O) is a non-redundant MTEC conjugate pair

We first need to prove some lemmas before proving this theorem. First, we need to study $\varepsilon_P(X)$ only on $[0, t_L[$ because $\varepsilon_P(X)$ is "L-periodic" on $[0, Nt_L[$.

Lemma A.2 (L-periodicity of $\varepsilon_P(X)$)**.** *Under the assumptions of Theorem 3.8, if $x \in \varepsilon_P(X)$ with $t_x < (N-2)t_L$, then $(x + L) \in \varepsilon_P(X)$.*

Remark. The condition $t_x < (N-2)t_L$ is necessary to stay in the definition domain of X when translating a point of $\varepsilon_P(X)$ by L.

Proof of Lemma A.2. Because $x \in \varepsilon_P(X)$, we have:

$$\begin{aligned}
& P_x \subseteq X \\
\Rightarrow\ & P_{x+L} \subseteq X_L && \text{(translation by } L\text{)} \\
\Rightarrow\ & P_{x+L_{[0,NL[}} \subseteq X_{L_{[0,NL[}} && \text{(by intersection)} \\
\Rightarrow\ & P_{x+L} \subseteq X_{L_{[0,NL[}} && (\forall p \in P, t_p + t_x + t_L < Nt_L) \\
\Rightarrow\ & P_{x+L} \subseteq (P \oplus O_L)_{[0,NL[} && \text{(definition of } X\text{)} \\
\Rightarrow\ & P_{x+L} \subseteq (P \oplus \{kL \mid k \in [\![1, N]\!]\})_{[0,NL[} && \text{(translation of } O \text{ by } L\text{)} \\
\Rightarrow\ & P_{x+L} \subseteq P \oplus \{kL \mid k \in [\![1, N-1]\!]\} && \text{(with the intersection)} \\
\Rightarrow\ & P_{x+L} \subseteq X && \text{(definition of } X\text{)}
\end{aligned}$$

Consequently $(x+L) \in \varepsilon_P(X)$, i.e. it is sufficient to study $\varepsilon_P(X)$ only in $[0, t_L[$. □

Lemma A.3 (Belonging to P_x)**.** *Under the assumptions of Theorem 3.8 and with $x \in \varepsilon_P(X)$, $t_x \in [0, t_L[$, and $p \in P$, we have: $t_x \leq t_p \Rightarrow p \in P_x$, and $t_x > t_p \Rightarrow p + L \in P_x$.*

Proof of Lemma A.3. Because $x \in \varepsilon_P(X)$, we have:

$$\begin{aligned}
& P_x \subseteq X \\
\Rightarrow\ & P_x \subseteq X_{[x,x+L[} \\
\Rightarrow\ & P_x \subseteq X_{[x,L[} \cup X_{[L,x+L[} \\
\Rightarrow\ & P_x \subseteq P_{[x,L[} \cup P_{L[L,x+L[}
\end{aligned}$$

However,

$$\begin{aligned}
|P_{[x,L[} \cup P_{L[L,x+L[}| &= |P_{[x,L[}| + |P_{L[L,x+L[}| \\
&= |P_{[x,L[}| + |P_{[0,x[}| \\
&= |P| \\
&= |P_x|
\end{aligned}$$

Because $P_x \subseteq P_{[x,L[} \cup P_{L[L,x+L[}$ and $|P_x| = |P_{[x,L[} \cup P_{L[L,x+L[}|$, we can conclude that $P_x = P_{[x,L[} \cup P_{L[L,x+L[}$, i.e. $P_{x[x,L[} = P_{[x,L[}$ and $P_{x[L,x+L[} = P_{L[L,x+L[}$.

$\boxed{t_x \leq t_p}$ Let $t_x \leq t_p$, therefore $p \in P_{[x,L[}$. Because $P_{x[x,L[} = P_{[x,L[}$, we have $p \in P_x$.

$\boxed{t_x > t_p}$ Let $t_x > t_p$, therefore $p \in P_{[0,x[}$. Translating by L, we have $p + L \in P_{L[L,x+L[}$. Because $P_{x[L,x+L[} = P_{L[L,x+L[}$, we have $p + L \in P_x$. □

According to the next lemma, it is enough to study $\varepsilon_P(X)$ on $[0, t_L/2]$ because there is a symmetry with respect to $t_L/2$.

Lemma A.4 (Symmetry of $\varepsilon_P(X)$ on $[0, t_L[$ with respect to $t_L/2$)**.** *Under the assumptions of Theorem 3.8, let $x \in \varepsilon_P(X)$ with $t_x \in [0, t_L[$, then $(L - x) \in \varepsilon_P(X)$.*

Proof of Lemma A.4. Let $p \in P$, we distinguish two cases: $t_x \leq t_p$ and $t_x > t_p$.
$\boxed{t_x \leq t_p}$ According to Lemma A.3, we have :

$$\begin{aligned} & p \in P_x & \\ \Rightarrow \quad & (p-x) \in P & \text{(translation by } -x) \\ \Rightarrow \quad & (p-x) \in X & (P \subseteq X) \\ \Rightarrow \quad & (p+L-x) \in X & (L\text{-periodicity of } X = P \oplus O) \end{aligned}$$

$\boxed{t_x > t_p}$ According to Lemma A.3, we have:

$$\begin{aligned} & (p+L) \in P_x & \\ \Rightarrow \quad & (p+L-x) \in P & \text{(translation by } -x) \\ \Rightarrow \quad & (p+L-x) \in X & (P \subseteq X) \end{aligned}$$

In all cases, $\forall p \in P, (p+L-x) \in X$ from which we conclude that $P_{L-x} \subseteq X$ therefore $(L-x) \in \varepsilon_P(X)$. □

Lemma A.5 ($\{0_E, x\}$ describes P)**.** *Under the assumptions of Theorem 3.8, let $x \in \varepsilon_P(X)$ with $t_x \in [0, t_L/2]$, then $\gamma_{\{0_E,x\}}(P) = P$, where 0_E denotes the origin of the set E.*

Proof of Lemma A.5. Let $p \in P$, we again distinguish two cases: $t_x \leq t_p$ and $t_p < t_x$.
$\boxed{t_x \leq t_p}$ According to Lemma A.3, $p \in P_x$. Thus $(p-x) \in P$.
$\boxed{t_x > t_p}$ Because $x \in \varepsilon_P(X)$, we have $(p+x) \in X$. However: $0 \leq t_p + t_x < 2t_x$, and $t_x \in [0, t_L/2]$, which leads to $0 \leq t_p + t_x < t_L$. We can conclude that $(p+x) \in P$.
Finally, $(p+x) \in P$ or $(p-x) \in P$, i.e. in any case: $\gamma_{\{0_E,x\}}(P) = P$. □

This provides the proof of Theorem 3.8.

Proof of Theorem 3.8. To prove that (P, O) is a non-redundant MTEC conjugate pair, we need to prove that $\varepsilon_P(X) = O$, $\varepsilon_O(X) = P$, and $|\gamma_P(X)| = |P| \times |O|$.

$\boxed{\varepsilon_P(X) = O}$ Let $x \in \varepsilon_P(X)$. According to Lemma A.2, we can suppose that $t_x \in [0, t_L[$ because $\varepsilon_P(X)$ is "L-periodic" on $[0, Nt_L[$. Moreover, with Lemma A.4 we can further reduce the interval by assuming $t_x \in [0, t_L/2]$ by symmetry of $\varepsilon_P(X)$ on $[0, t_L[$ with respect to $t_L/2$. In this case, Lemma A.5 allows us to conclude that $\gamma_{\{0_E,x\}}(P) = P$. However, P is non-periodic, which leads to $x = 0_E$. With the symmetry and the $L-$periodicity of $\varepsilon_P(X)$, this proves that $\varepsilon_P(X) = O$.

$\boxed{\varepsilon_O(X) = P}$ We use Theorem 3.2. X is bounded by definition and $\gamma_P(X) = X$. Moreover, $\forall p \in E : P_p \subseteq X \Rightarrow p = kL$ with $k \in [\![0, N-1]\!]$ because $\varepsilon_P(X) = O$. By definition of X, $\mathrm{Ch}(P_{kL}) \cap X = P_{kL}$ from which we conclude that P is full in X. Applying Theorem 3.2, we obtain $\varepsilon_O(X) = P$.

$\boxed{|\gamma_P(X)| = |P| \times |O|}$ This equality is due to the fact that the occurrences of the pattern P do not intersect in X because P is of temporal length less than t_L. □

Appendix B

Bayesian Model for Modeling Boundary Credence in Music Performance

---

We describe here the method for estimating boundary credences from tempo or loudness data of a musical performance. We first introduce the notations employed here, present our adaptation of the forward-backward algorithm, explain the arc model, and describe the priors we have chosen for our model.

B.1 Problem Statement and Notations

The notations used are listed here. We consider that the performance segmentation consists of a succession of non-overlapping, consecutive intervals that can only change on the beat.

- *Prosodic feature* $\mathcal{D}$: extracted from the recorded performance as a sequence of N instantaneous tempo or loudness values, each corresponding to the value at a beat (this is coherent with the MazurkaBL dataset [Kosta 2018] as the information is available at beat level).
- *Sequence slice* $\mathcal{D}\,[i, j]$: musical prosody parameter values from indices i to and including j (0-indexed).
- *Performance segmentation* S: set of integer intervals which represent the segmentation as a succession of non-overlapping consecutive intervals.
- *Arc* $[i, j] \in S$: starting at index i and ending at index j, and the next arc (if there is one) would start at index $j + 1$.
- *Arc starting at* i $[i, \sim] \in S$: regardless of its other end (i.e. $[i, \sim] \in S \iff \exists j : [i, j] \in S$), respectively *an arc ending at* j (i.e. $[\sim, j] \in S \iff \exists i : [i, j] \in S$).
- *Posterior boundary credences* $p([\sim, j] \in S \mid \mathcal{D})$ or *posterior arc credences* $p([i, j] \in S \mid \mathcal{D})$: Probability that a segment ends at j, or probability that a segment starts at i and ends at j. The aim of the method is to compute them.
- *Boundary credence profile* and *arc credence matrix*: output of the method, although the matrix will be further transformed for the sake of visualisation.

The assumption of independence across arcs is formalized in two ways, one for the data and one for the prior:

$$\forall n, \quad \mathcal{D}\left[0, n\right] \perp\!\!\!\perp \mathcal{D}\left[n+1, N-1\right] \mid [\sim, n] \in S \tag{B.1}$$

$$\forall k, l, i, j : k < l < i < j, \quad p([i, j] \in S \mid [i, \sim] \in S, [k, l] \in S) = \lambda\left(i, j\right) \tag{B.2}$$

where $\lambda\left(i, j\right) = p([i, j] \in S \mid [i, \sim] \in S)$. In less formal terms, this means that the data before a boundary has no effect on the data after that boundary and that the prior on the end of a segment does not depend on previous segments.

Finally, we assume that the first and last beats are respectively the first and the last beats of the corresponding arcs, i.e. $[0, \sim] \in S$ and $[\sim, N-1] \in S$. The λ function is then sufficient to define the entire prior on segmentations[1] and operates as a parameter for the method. We use functions that are translation invariant, meaning they act as a prior on the length of any segment, but this is not a requirement.

B.2 Adapted Forward-Backward Algorithm

We demonstrate here that the posterior marginals can be computed efficiently, by using a similar process to that of the forward-backward algorithm, which bears resemblance to some Bayesian changepoint detection algorithms [Fearnhead 2011, Rigaill 2012].

By applying Bayes' formula, we have the following equation:

$$p([\sim, n] \in S \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid [\sim, n] \in S) p([\sim, n] \in S)}{p(\mathcal{D})} \tag{B.3}$$

None of these terms are trivial to compute. However, using the assumption that data across arcs is independent, we can rewrite Equation B.3 using so-called forward and backward quantities $\alpha(n)$ and $\beta(n)$, defined as[2]:

$$\forall n \in \{0, \ldots, N-1\}, \qquad \alpha(n) = p(\mathcal{D}\left[0, n\right], [\sim, n] \in S) \tag{B.4}$$

$$\forall n \in \{-1, \ldots, N-2\}, \qquad \beta(n) = p(\mathcal{D}\left[n+1, N-1\right] \mid [n+1, \sim] \in S) \tag{B.5}$$

$$\forall n \in \{0, \ldots, N-1\}, \qquad p([\sim, n] \in S \mid \mathcal{D}) = \frac{\alpha(n)\beta(n)}{\alpha(N-1)} \tag{B.6}$$

which in very rough terms split the probability of observing the overall data according to a hypothetical boundary at n. An important difference is that α is a joint probability while β is a conditional probability.

[1] It is possible to go from this λ function to the implied prior credence on boundaries and segments by applying the same algorithm while neutralizing the data likelihood terms.

[2] Note that $\alpha(N-1) = \beta(-1) = p(\mathcal{D})$. We also extend the definitions such that $\alpha(-1) = \beta(N-1) = 1$ to handle boundary conditions.

Recursive formulae can be derived for these new quantities, using $\kappa(i,j) = p(\mathcal{D}[i,j] \mid [i,j] \in S)$:

$$\alpha(n) = \sum_{i=0}^{n-1} \alpha(i-1) \times \lambda(i,n) \times \kappa(i,n) \tag{B.7}$$

$$\beta(n) = \sum_{i=n+2}^{N-1} \beta(i) \times \lambda(n+1,i) \times \kappa(n+1,i) \tag{B.8}$$

showing that both quantities can be computed respectively forward and backward by summing, over possible arcs, their previously computed values, weighted by the prior on that arc and the likelihood of the corresponding data slice. This can be done efficiently using dynamic programming, especially if the prior is null for arcs over a maximum length. Specifically, in $\mathcal{O}(NK)$ if K is the maximum arc length.

Additionally, we can once again use the independence of data across arcs to get posterior marginals on each arc:

$$p([i,j] \in S \mid \mathcal{D}) = \frac{\alpha(i)\kappa(i,j)\lambda(i,j)\beta(j)}{\alpha(N-1)} \tag{B.9}$$

Provided we have a specific model and algorithm which can yield the $\kappa(i,j)$ for all relevant pairs, we can thus compute efficiently the posterior credences. The next section describes one such model.

B.3 Arc Model

The arc-level model is a fairly standard Bayesian polynomial model, as can be found in [Bishop 2006], from which notations are mostly borrowed. The main difference in the approach is that we are not ultimately interested in the model parameters, but in the likelihood of the data segment.

Throughout this section, we work under the assumption that there is an arc from index i to index j, with $j > i$. To insulate the arc model from the global considerations, we pose $\mathbf{t} = \mathcal{D}[i,j]$ and we define $\mathbf{x}$ to be the normalized score time within the arc, formally:

$$\mathbf{x} = \left(\frac{k}{j-i}\right)_{k \in \{0,\dots,j-i\}} \tag{B.10}$$

All variables defined in this section are then to be taken with respect to i and j, apart from the prior parameters μ, Σ and η, which are constant across all arcs.

First we assume that there is an *ideal* tempo series $\mathbf{y}$, representing the performer's intended tempo curve for the phrase, from which the observed data $\mathbf{t}$ deviates by ε. This deviation term is meant to encapsulate a variety of sources, such as finer-scale modulation (e.g. note-level *rubato*), beat annotation/extraction error, execution error, etc. and is modelled as independent, centered Gaussian noise, with a fixed variance η:

$$\mathbf{t} = \mathbf{y} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \eta I) \tag{B.11}$$

where I is the identity matrix of the appropriate size.

We then model the ideal tempo as a quadratic function of score time as described in Section 7.2.1, with independent Gaussian priors on its parameters:

$$\mathbf{y} = \Phi_{\mathbf{x}}\mathbf{w}, \qquad \mathbf{w} \sim \mathcal{N}(\mu, \Sigma)\,, \qquad \Phi_{\mathbf{x}} := \begin{bmatrix} x_0^2 & x_0 & 1 \\ \vdots & \vdots & \vdots \\ x_{N-1}^2 & x_{N-1} & 1 \end{bmatrix}, \tag{B.12}$$

where the $x_0, \ldots, x_{N-1}$ are the individual values of $\mathbf{x}$, and μ and Σ hold respectively the means and (diagonal) covariance matrix of the priors on the quadratic, linear and constant coefficients (in that order). In sum, we have:

$$p(\mathbf{t} \mid \mathbf{w}) = \mathcal{N}(\mathbf{t}|\Phi_{\mathbf{x}}\mathbf{w}, \eta I)\,, \tag{B.13}$$

which, marginalising against $\mathbf{w}$, yields:

$$p(\mathbf{t}) = \mathcal{N}\Big(\mathbf{t}|\Phi_{\mathbf{x}}\mu, \eta I + \Phi_{\mathbf{x}}\Sigma{\Phi_{\mathbf{x}}}^{\mathsf{T}}\Big) \tag{B.14}$$

B.4 Prior Setup

As always with Bayesian methods, the output is dependent on the priors. For the method, we need to select priors for likely segment lengths, phrase arc parameters, and noise.

Tempo and loudness arc priors: In order to set reasonable priors, tempo and loudness arc boundaries were manually annotated for 37 performances across 4 pieces. Maximum likelihood estimates were then fitted to each arc in order to infer the corresponding model parameters, whose mean and variance were then used to construct the different priors. The resulting prior parameters were:

$$\mu_{tempo} = \begin{pmatrix} -181 \\ 159 \\ 107 \end{pmatrix}, \quad \Sigma_{tempo} = \begin{bmatrix} 93^2 & 0 & 0 \\ 0 & 106^2 & 0 \\ 0 & 0 & 31^2 \end{bmatrix}, \quad \eta_{tempo} = 18.1 \tag{B.15}$$

$$\mu_{loud} = \begin{pmatrix} -0.73 \\ 0.68 \\ 0.41 \end{pmatrix}, \quad \Sigma_{loud} = \begin{bmatrix} 0.55^2 & 0 & 0 \\ 0 & 0.60^2 & 0 \\ 0 & 0 & 0.19^2 \end{bmatrix}, \quad \eta_{loud} = 0.039 \tag{B.16}$$

Segment length priors: For the prior on segment length, we have used a discretised Gaussian distribution, cut off at 30 beats, with mean 14.7 and standard deviation 5.95 (again set according to the 37 manual annotations).

These priors are wide, which is expected as the arcs can exhibit very different shapes and expectations; priors that are too strict would likely result in poor segmentations. Overall, this means that posterior credences are mainly driven by the goodness of the arc fits.

Bibliography

[Adams 1976] Charles R. Adams. *Melodic Contour Typology.* Ethnomusicology, vol. 20, no. 2, pages 179–215, 1976. (Cited on page 92.)

[Agon 2018] Carlos Agon, Moreno Andreatta, Jamal Atif, Isabelle Bloch and Pierre Mascarade. *Musical Descriptions Based on Formal Concept Analysis and Mathematical Morphology.* In International Conference on Conceptual Structures, pages 105–119. Springer, 2018. (Cited on page 12.)

[Anagnostopoulou 2013] Christina Anagnostopoulou, Mathieu Giraud and Nick Poulakis. *Melodic Contour Representations in the Analysis of Children's Songs.* In 3rd International Workshop on Folk Music Analysis, pages 40–43, Amsterdam, Netherlands, 2013. (Cited on page 92.)

[Bååth 2014] Rasmus Bååth, Erik Lagerstedt and Peter Gärdenfors. *A Prototype-Based Resonance Model of Rhythm Categorization.* i-Perception, vol. 5, no. 6, pages 548–558, 2014. (Cited on page 129.)

[Bishop 2006] Christopher M. Bishop. Pattern Recognition and Machine Learning. Springer, New York, NY, USA, 2006. (Cited on page 191.)

[Bloch 2007] Isabelle Bloch, Henk Heijmans and Christian Ronse. Mathematical morphology, pages 857–944. Springer Netherlands, Dordrecht, 2007. (Cited on pages 13 and 89.)

[Buteau 2008] Chantal Buteau and Guerino Mazzola. *Motivic Analysis According to Rudolph Réti: Formalization by a Topological Model.* Journal of Mathematics and Music, vol. 2, no. 3, pages 117–134, 2008. (Cited on page 92.)

[Cambouropoulos 1996] Emilios Cambouropoulos. *A General Pitch Interval Representation: Theory and Applications.* Journal of New Music Research, vol. 25, no. 3, pages 231–251, 1996. (Cited on page 112.)

[Cambouropoulos 2006] Emilios Cambouropoulos. *Musical Parallelism and Melodic Segmentation: a Computational Approach.* Music Perception, vol. 23, no. 3, pages 249–268, 2006. (Cited on pages 5, 83 and 105.)

[Cancino-Chacón 2018] Carlos Eduardo Cancino-Chacón, Maarten Grachten, Werner Goebl and Gerhard Widmer. *Computational Models of Expressive Music Performance: A Comprehensive and Critical Review.* Frontiers in Digital Humanities, vol. 5, pages 5–25, 2018. (Cited on pages 5, 127 and 136.)

[Chew 2001] Elaine Chew. *Modeling Tonality: Applications to Music Cognition.* In 23rd Annual Meeting of the Cognitive Science Society, pages 206–211, 2001. (Cited on page 117.)

[Chew 2002] Elaine Chew. *The Spiral Array: an Algorithm for Determining Key Boundaries.* In International Conference on Music and Artificial Intelligence, pages 18–31. Springer, 2002. (Cited on page 104.)

[Chew 2016] Elaine Chew. *Playing with the Edge: Tipping Points and the Role of Tonality.* Music Perception, vol. 33, no. 3, pages 344–366, 2016. (Cited on page 135.)

[Chew 2021] Elaine Chew, Michele Orini and Pier Lambiase. *Putting (One's) Heart into Music.* European Heart Journal, vol. 42, no. 28, pages 2721–2724, 2021. (Cited on page 142.)

[Chizat 2018] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer and François-Xavier Vialard. *Scaling Algorithms for Unbalanced Optimal Transport Problems.* Mathematics of Computation, vol. 87, no. 314, pages 2563–2609, 2018. (Cited on page 157.)

[Chuan 2007] Ching Hua Chuan and Elaine Chew. *A Dynamic Programming Approach to the Extraction of Phrase Boundaries from Tempo Variations in Expressive Performances.* In 8th International Society for Music Information Retrieval Conference, pages 305–308, Vienna, Austria, 2007. (Cited on pages 127 and 147.)

[Cohn 1997] Richard Cohn. *Neo-Riemannian Operations, Parsimonious Trichords, and Their "Tonnetz" Representations.* Journal of Music Theory, vol. 41, no. 1, pages 1–66, 1997. (Cited on page 3.)

[Collins 2010] Tom Collins, Jeremy Thurlow, Robin Laney, Alistair Willis and Paul Garthwaite. *A Comparative Evaluation of Algorithms for Discovering Translational Patterns in Baroque Keyboard Works.* In 11th International Society for Music Information Retrieval Conference, Utrecht, Netherlands, pages 3–8, 2010. (Cited on page 36.)

[Collins 2011] Tom Collins. *Improved Methods for Pattern Discovery in Music, with Applications in Automated Stylistic Composition.* PhD thesis, Faculty of Mathematics, Computing and Technology, The Open University, United Kingdom, 2011. (Cited on pages 36, 62, 63 and 67.)

[Collins 2013] Tom Collins and David Meredith. *Maximal Translational Equivalence Classes of Musical Patterns in Point-Set Representations.* In 4th International Conference on Mathematics and Computation in Music, Montreal, QC, Canada, pages 88–99, 2013. (Cited on pages 42, 43, 44, 49, 54 and 67.)

[Cook 2007] Nicholas Cook. *Performance Analysis and Chopin's Mazurkas.* Musicae Scientiae, vol. 11, no. 2, pages 183–207, 2007. (Cited on pages 127 and 159.)

[Cook 2014] Nicholas Cook. *Between Art and Science: Music as Performance.* Journal of the British Academy, vol. 2, pages 1–25, 2014. (Cited on page 127.)

[De Haas 2016] W. Bas De Haas and Anja Volk. *Meter Detection in Symbolic Music Using Inner Metric Analysis.* In 17th International Society for Music Information Retrieval Conference, New York City, USA, pages 441–447, 2016. (Cited on page 82.)

[Desain 2003] Peter Desain and Henkjan Honing. *The Formation of Rhythmic Categories and Metric Priming.* Perception, vol. 32, no. 3, pages 341–365, 2003. (Cited on pages 128 and 129.)

[Douthett 1998] Jack Douthett and Peter Steinbach. *Parsimonious Graphs: a Study in Parsimony, Contextual Transformations, and Modes of Limited Transposition.* Journal of Music Theory, pages 241–263, 1998. (Cited on pages 3 and 115.)

[Dowling 1994] W. Jay Dowling. *Melodic Contour in Hearing and Remembering Melodies.* In Musical Perceptions, page 173–190. Oxford University Press, 1994. (Cited on page 92.)

[Euler 1739] Leonhard Euler. Tentamen Novae Theoriae Musicae. Wentworth Press (2019), 1739. (Cited on page 3.)

[Fastl 2005] Hugo Fastl. *Psycho-Acoustics and Sound Quality.* In Communication acoustics, pages 139–162. Springer, 2005. (Cited on page 147.)

[Fearnhead 2011] Paul Fearnhead and Zhen Liu. *Efficient Bayesian Analysis of Multiple Changepoint Models With Dependence Across Segments.* Statistics and Computing, vol. 21, no. 2, pages 217–229, 2011. (Cited on page 190.)

[Foote 1999] Jonathan Foote. *Visualizing Music and Audio Using Self-similarity.* In 7th ACM International Conference on Multimedia (Part 1), Orlando, FL, USA, pages 77–80, 1999. (Cited on pages 94, 97 and 105.)

[Foote 2000] Jonathan Foote. *Automatic Audio Segmentation Using a Measure of Audio Novelty.* In IEEE International Conference on Multimedia and Expo ICME Computer Society Press, volume 1, pages 452–455, 2000. (Cited on page 98.)

[Forth 2009] Jamie Forth and Geraint A. Wiggins. *An Approach for Identifying Salient Repetition in Multidimensional Representations of Polyphonic Music.* London Algorithmics 2008: Theory and Practice, pages 44–58, 2009. (Cited on pages 39 and 54.)

[Forth 2012] Jamie Forth. *Cognitively-Motivated Geometric Methods of Pattern Discovery and Models of Similarity in Music.* PhD thesis, Goldsmiths, University of London, 2012. (Cited on page 39.)

[Foscarin 2020] Francesco Foscarin, Andrew McLeod, Philippe Rigaux, Florent Jacquemard and Masahiko Sakai. *ASAP: A Dataset of Aligned Scores and Performances for Piano Transcription.* In 21st International Society for Music Information Retrieval Conference, pages 534–541, 2020. (Cited on page 160.)

[Gabrielsson 1987] Alf Gabrielsson. *Once Again: The Theme from Mozart's Piano Sonata in A Major.* Action and Perception in Rhythm and Music, pages 81–103, 1987. (Cited on pages 127, 147 and 149.)

[Ghias 1995] Asif Ghias, Jonathan Logan, David Chamberlin and Brian C Smith. *Query by Humming: Musical Information Retrieval in an Audio Database.* In 3rd ACM International Conference on Multimedia, pages 231–236, 1995. (Cited on page 113.)

[Giraud 2012a] Mathieu Giraud, Richard Groult and Florence Levé. *Detecting Episodes with Harmonic Sequences for Fugue Analysis.* In 13th International Society for Music Information Retrieval Conference, Porto, Portugal, 2012. (Cited on page 30.)

[Giraud 2012b] Mathieu Giraud, Richard Groult and Florence Levé. *Subject and Counter-Subject Detection for Analysis of the Well-Tempered Clavier Fugues.* In International Symposium on Computer Music Modeling and Retrieval, pages 422–438. Springer, 2012. (Cited on pages 30 and 112.)

[Giraud 2015] Mathieu Giraud, Richard Groult, Emmanuel Leguy and Florence Levé. *Computational Fugue Analysis.* Computer Music Journal, vol. 39, no. 2, pages 77–96, 2015. (Cited on pages 30 and 113.)

[Giraud 2016] Mathieu Giraud, Richard Groult and Florence Levé. *Computational Analysis of Musical Form.* In Computational Music Analysis, pages 113–136. Springer, 2016. (Cited on page 104.)

[Gotham 2015] Mark Gotham. *The Metre Metrics: Characterising (Dis)Similarity Among Metrical Structures.* PhD thesis, University of Cambridge, 2015. (Cited on pages 167 and 168.)

[Gromov 1999] Mikhael Gromov, Misha Katz, Pierre Pansu and Stephen Semmes. Metric Structures for Riemannian and Non-Riemannian Spaces, volume 152. Springer, 1999. (Cited on page 157.)

[Guichaoua 2024] Corentin Guichaoua, Paul Lascabettes and Elaine Chew. *End-To-End Bayesian Segmentation and Similarity Assessment of Performed Music Tempo and Dynamics With No Score Information.* Music & Science, 2024. (Cited on page 6.)

[Gusfield 1997] Dan Gusfield. Algorithms on Stings, Trees, and Sequences: Computer Science and Computational Biology. Cambridge University Press, 1997. (Cited on page 30.)

[Hadwiger 1950] Hugo Hadwiger. *Minkowskische Addition und Subtraktion beliebiger Punkt-mengen und die Theoreme von Erhard Schmidt.* Mathematische Zeitschrift, vol. 53, pages 210–218, 1950. (Cited on pages 13 and 90.)

[Handel 1993] Stephen Handel. *The Effect of Tempo and Tone Duration on Rhythm Discrimination.* Perception & Psychophysics, vol. 54, no. 3, 1993. (Cited on page 129.)

[Hanenberg 2018] Scott James Hanenberg. *Unpopular Meters Irregular Grooves and Drumbeats in the Songs of Tori Amos, Radiohead, and Tool.* PhD thesis, University of Toronto, 2018. (Cited on page 167.)

[Haralick 1985] Robert M. Haralick and Linda G. Shapiro. *Image Segmentation Techniques.* Computer Vision, Graphics, and Image Processing, vol. 29, no. 1, pages 100–132, 1985. (Cited on page 147.)

[Heijmans 1990] Henk J.A.M Heijmans and Christian Ronse. *The Algebraic Basis of Mathematical Morphology: I. Dilations and Erosions.* Computer Vision, Graphics, and Image Processing, vol. 50, no. 3, pages 245–295, 1990. (Cited on pages 13 and 89.)

[Heijmans 1994] Henk J.A.M. Heijmans. Morphological Image Operators. Academic Press, 1994. (Cited on pages 13 and 89.)

[Herremans 2016] Dorien Herremans and Elaine Chew. *Tension Ribbons: Quantifying and Visualising Tonal Tension.* In 2nd International Conference on Technologies for Music Notation and Representation, page 10, Cambridge, United Kingdom, 2016. (Cited on pages 136 and 153.)

[Honing 2002] Henkjan Honing. *Structure and Interpretation of Rhythm and Timing.* Tijdschrift voor Muziektheorie, vol. 7, no. 3, pages 227–232, 2002. (Cited on page 128.)

[Hsu 1998] Jia-Lien Hsu, Arbee L.P. Chen and Chih-Chin Liu. *Efficient Repeating Pattern Finding in Music Databases.* In 7th International Conference on Information and Knowledge Management, pages 281–288, 1998. (Cited on pages 30 and 105.)

[Hsu 2001] Jia-Lien Hsu, Chih-Chin Liu and Arbee L.P. Chen. *Discovering Nontrivial Repeating Patterns in Music Data.* IEEE Transactions on Multimedia, vol. 3, no. 3, pages 311–325, 2001. (Cited on page 105.)

[Jacoby 2017] Nori Jacoby and Josh H. McDermott. *Integer Ratio Priors on Musical Rhythm Revealed Cross-Culturally by Iterated Reproduction.* Current Biology, vol. 27, no. 3, pages 359–370, 2017. (Cited on pages 129 and 142.)

[Janssen 2013] Berit Janssen, W. Bas De Haas, Anja Volk and Peter Van Kranenburg. *Discovering Repeated Patterns in Music: State of Knowledge, Challenges, Perspectives.* In 10th International Symposium on Computer Music Multidisciplinary Research, Marseille, France, volume 20, page 74, 2013. (Cited on pages 30 and 71.)

[Kantorovich 1942] Leonid V. Kantorovich. *On the Translocation of Masses.* In Doklady Akademii Nauk, volume 37, pages 199–201, 1942. (Cited on page 155.)

[Karvonen 2008] Mikko Karvonen and Kjell Lemström. *Using Mathematical Morphology for Geometric Music Information Retrieval.* In International Workshop on Machine Learning and Music, 2008. (Cited on pages 12, 72 and 80.)

[Karvonen 2010] Mikko Karvonen, Mika Laitinen, Kjell Lemström and Juho Vikman. *Error-Tolerant Content-Based Music-Retrieval with Mathematical Morphology.* In International Symposium on Computer Music Modeling and Retrieval, volume LNCS 6684, pages 321–337, 2010. (Cited on pages 12, 72 and 80.)

[Koprinska 2001] Irena Koprinska and Sergio Carrato. *Temporal Video Segmentation: A Survey.* Signal Processing: Image Communication, vol. 16, no. 5, pages 477–500, 2001. (Cited on page 147.)

[Kosta 2018] Katerina Kosta, Oscar F. Bandtlow and Elaine Chew. *MazurkaBL: Score-Aligned Loudness, Beat, Expressive Markings Data for 2000 Chopin Mazurka Recordings.* In 4th International Conference on Technologies for Music Notation and Representation, pages 85–94, Montreal, Canada, 2018. (Cited on pages 6, 127, 129, 134, 148, 160 and 189.)

[Krumhansl 1998] Carol L. Krumhansl. *Perceived Triad Distance: Evidence Supporting the Psychological Reality of Neo-Riemannian Transformations.* Journal of Music Theory, vol. 42, no. 2, pages 265–281, 1998. (Cited on page 115.)

[Langner 2003] Jörg Langner and Werner Goebl. *Visualizing Expressive Performance in Tempo-Loudness Space.* Computer Music Journal, vol. 27, no. 4, pages 69–83, 2003. (Cited on page 127.)

[Lascabettes 2019] Paul Lascabettes. *Mathematical Morphology Applied to Music.* Master's thesis, École Normale Supérieure Paris-Saclay, 2019. (Cited on pages 12 and 19.)

[Lascabettes 2020] Paul Lascabettes, Isabelle Bloch and Carlos Agon. *Analyse de représentations spatiales de la musique par des opérateurs simples de morphologie mathématique.* In Journées d'Informatique Musicale, Strasbourg, France, 2020. (Cited on pages 7, 12 and 19.)

[Lascabettes 2022a] Paul Lascabettes, Carlos Agon, Moreno Andreatta, and Isabelle Bloch. *Computational Analysis of Musical Structures based on Morphological Filters.* In 9th International Conference on Mathematics and Computation in Music, pages 267–278, Atlanta, GA, USA, 2022. (Cited on pages 7, 12 and 88.)

[Lascabettes 2022b] Paul Lascabettes, Corentin Guichaoua and Elaine Chew. *Generating Multiple Hierarchical Segmentations of Music Sequences Using Adapted Correlative Matrices.* In 19th Sound and Music Computing Conference, pages 338–345, Saint Etienne, France, 2022. (Cited on pages 6 and 104.)

[Lascabettes 2023] Paul Lascabettes, Elaine Chew and Isabelle Bloch. *Characterizing and Interpreting Music Expressivity through Rhythm and Loudness Simplices.* In International Computer Music Conference, Shenzhen, China, 2023. (Cited on pages 6 and 126.)

[Lascabettes 2024] Paul Lascabettes and Isabelle Bloch. *Discovering Repeated Patterns From the Onsets in a Multidimensional Representation of Music.* In International Conference on Discrete Geometry and Mathematical Morphology, Florence, Italie, 2024. (Cited on page 6.)

[Le Moine 2020] Clément Le Moine and Nicolas Obin. *Att-HACK: An Expressive Speech Database with Social Attitudes.* In Speech Prosody 2020, pages 744–748, 2020. (Cited on page 140.)

[Lerdahl 1985] Fred Lerdahl and Ray Jackendoff. A Generative Theory of Tonal Music. MIT Press, 1985. (Cited on pages 60, 72, 104, 106, 110, 166 and 167.)

[Lerdahl 2001] Fred Lerdahl. Tonal Pitch Space. Oxford University Press, USA, 2001. (Cited on page 104.)

[Lindenmayer 1968] Aristid Lindenmayer. *Mathematical Models for Cellular Interactions in Development I. Filaments with One-Sided Inputs.* Journal of Theoretical Biology, vol. 18, no. 3, pages 280–299, 1968. (Cited on page 129.)

[London 2012] Justin London. Hearing in Time: Psychological Aspects of Musical Meter. Oxford University Press, 2012. (Cited on page 167.)

[Lu 2004] Lie Lu, Muyuan Wang and Hong-Jiang Zhang. *Repeating Pattern Discovery and Structure Analysis From Acoustic Music Data.* In 6th ACM SIGMM International Workshop on Multimedia Information Retrieval, pages 275–282, 2004. (Cited on page 89.)

[Mahalanobis 1936] Prasanta Chandra Mahalanobis. *On the Generalised Distance in Statistics*. National Institute of Sciences of India, vol. 2, no. 1, page 49–55, 1936. (Cited on page 138.)

[Mantel 1967] Nathan Mantel. *The Detection of Disease Clustering and a Generalized Regression Approach*. Cancer research, vol. 27, no. 2 Part 1, pages 209–220, 1967. (Cited on page 160.)

[Marvin 1987] Elizabeth West Marvin and Paul A. Laprade. *Relating Musical Contours: Extensions of a Theory for Contour*. Journal of Music Theory, vol. 31, no. 2, pages 225–267, 1987. (Cited on page 93.)

[Matheron 2002] Georges Matheron and Jean Serra. *The Birth of Mathematical Morphology*. In 6th International Symposium on Mathematical Morphology, pages 1–16. Sydney, Australia, 2002. (Cited on page 12.)

[McLeod 2017] Andrew McLeod and Mark Steedman. *Meter Detection in Symbolic Music Using a Lexicalized PCFG*. In 14th Sound and Music Computing Conference, pages 373–379, 2017. (Cited on page 82.)

[Meredith 1999] David Meredith. *The Computational Representation of Octave Equivalence in the Western Staff Notation System*. In Cambridge Music Processing Colloquium, 1999. (Cited on page 112.)

[Meredith 2001] David Meredith, Geraint A. Wiggins and Kjell Lemström. *Pattern Induction and Matching in Polyphonic Music and Other Multidimensional Datasets*. In 5th World Multiconference on Systemics, Cybernetics and Informatics, volume 10, pages 61–66, 2001. (Cited on pages 32, 33, 40 and 63.)

[Meredith 2002a] David Meredith, Kjell Lemström and Geraint A. Wiggins. *Algorithms for Discovering Repeated Patterns in Multidimensional Representations of Polyphonic Music*. Journal of New Music Research, vol. 31, no. 4, pages 321–345, 2002. (Cited on pages 4, 31, 32, 33, 36, 37, 40, 41, 47, 54, 62, 63, 70, 72 and 112.)

[Meredith 2002b] David Meredith, Geraint Wiggins and Kjell Lemström. *Method of Pattern Discovery*, 2002. GB2379056. (Cited on page 40.)

[Meredith 2006] David Meredith. *Point-Set Algorithms for Pattern Discovery and Pattern Matching in Music*. In Dagstuhl Seminar Proceedings. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2006. (Cited on pages 33, 37, 63 and 70.)

[Meredith 2013] David Meredith. *COSIATEC and SIATECCompress: Pattern Discovery by Geometric Compression*. In 14th International Society for Music Information Retrieval Conference, 2013. (Cited on pages 37, 38, 39, 54 and 70.)

[Meredith 2015] David Meredith. *Music Analysis and Point-Set Compression*. Journal of New Music Research, vol. 44, no. 3, pages 245–270, 2015. (Cited on pages 39 and 40.)

[Meredith 2016] David Meredith. *Using SIATECCompress to Discover Repeated Themes and Sections in Polyphonic Music*. In Music Information Retrieval Evaluation Exchange, 2016. (Cited on pages 67 and 83.)

[Minkowski 1903] Hermann Minkowski. *Volumen und Oberfläche*. Mathematische Annalen, vol. 57, pages 447–495, 1903. (Cited on pages 13 and 90.)

[Monge 1781] Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais*. Imprimerie Royale, pages 666–704, 1781. (Cited on page 155.)

[Najman 2010] Laurent Najman and Hugues Talbot. Mathematical Morphology: From Theory to Applications. ISTE-Wiley, 2010. (Cited on pages 13 and 89.)

[Nave 2021] Karli Nave, Chantal Carrillo, Nori Jacoby, Laurel Trainor and Erin Hannon. *The Development of Rhythmic Categories as Revealed Through an Iterative Production Task*. Unpublished, 2021. (Cited on page 129.)

[Nieto 2020] Oriol Nieto, Gautham J. Mysore, Cheng-i Wang, Jordan B.L. Smith, Jan Schlüter, Thomas Grill and Brian McFee. *Audio-Based Music Structure Analysis: Current Trends, Open Challenges, and Applications*. Transactions of the International Society for Music Information Retrieval, vol. 3, no. 1, pages 246–263, 2020. (Cited on pages 147 and 148.)

[Palmer 2006] Caroline Palmer and Sean Hutchins. *What Is Musical Prosody?* Psychology of Learning and Motivation, vol. 46, 2006. (Cited on page 147.)

[Patricio 2014] Carlos Vaquero Patricio and Henkjan Honing. *Generating Expressive Timing by Combining Rhythmic Categories and Lindenmayer Systems*. In 50th Anniversary Convention of the AISB, pages 1–7, London, UK, 2014. (Cited on page 129.)

[Paulus 2010] Jouni Paulus, Meinard Müller and Anssi Klapuri. *State of the Art Report: Audio-Based Music Structure Analysis*. In 11th International Society for Music Information Retrieval Conference, pages 625–636. Utrecht, 2010. (Cited on pages 5, 94, 100, 101 and 147.)

[Pedregosa 2011] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg*et al. Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011. (Cited on page 159.)

[Pieslak 2007] Jonathan Pieslak. *Re-Casting Metal: Rhythm and Meter in the Music of Meshuggah.* Music Theory Spectrum, vol. 29, no. 2, pages 219–245, 2007. (Cited on pages 81 and 170.)

[Quinn 1999] Ian Quinn. *The Combinatorial Model of Pitch Contour.* Music Perception: An Interdisciplinary Journal, vol. 16, no. 4, pages 439–456, 1999. (Cited on page 93.)

[Rafael 2010] Brigitte Rafael and Stefan M. Oertl. *MTSSM–A Framework for Multi-Track Segmentation of Symbolic Music.* International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol. 4, no. 1, pages 7–13, 2010. (Cited on pages 104, 105 and 113.)

[Rafailidis 2008] Dimitris Rafailidis, Alexandros Nanopoulos, Yannis Manolopoulos and Emilios Cambouropoulos. *Detection of Stream Segments in Symbolic Musical Data.* In 9th International Conference on Music Information Retrieval, pages 83–88, 2008. (Cited on page 30.)

[Rigaill 2012] Guillem Rigaill, Emilie Lebarbier and Stéphane Robin. *Exact Posterior Distributions and Model Selection Criteria for Multiple Change-Point Detection Problems.* Statistics and Computing, vol. 22, no. 4, pages 917–929, 2012. (Cited on page 190.)

[Rink 1995] John Rink. *Playing in Time: Rhythm, Metre and Tempo in Brahms's Fantasien Op. 116.* The Practice of Performance: Studies in Musical Interpretation, pages 254–282, 1995. (Cited on page 146.)

[Romero-García 2022a] Gonzalo Romero-García, Isabelle Bloch and Carlos Agon. *Mathematical Morphology Operators for Harmonic Analysis.* In 9th International Conference on Mathematics and Computation in Music, pages 255–266, 2022. (Cited on pages 12 and 19.)

[Romero-García 2022b] Gonzalo Romero-García, Paul Lascabettes and Elaine Chew. *Automated Musical Rhythm Transcription of ECG RR Interval Time Series as a Tool for Representing Rhythm Variations and Annotation Anomalies in Arrhythmia Heartbeat Classifications.* In Computing in Cardiology, volume 498, 2022. (Cited on pages 7 and 142.)

[Ronse 1991] Christian Ronse and Henk J.A.M. Heijmans. *The Algebraic Basis of Mathematical Morphology: II. Openings and Closings.* CVGIP: Image Understanding, vol. 54, no. 1, pages 74–97, 1991. (Cited on pages 13 and 89.)

[Rupprecht 2017] Christian Rupprecht, Iro Laina, Robert Dipietro, Maximilian Baust, Federico Tombari, Nassir Navab and Gregory D. Hager. *Learning in an Uncertain World: Representing Ambiguity Through Multiple Hypotheses.* In IEEE International Conference on Computer Vision, pages 3611–3620, Venice, Italy, 2017. (Cited on page 148.)

[Sakran 2017] Alaa Ehab Sakran, Sherif Mahdy Abdou, Salah Eldeen Hamid and Mohsen Rashwan. *A Review: Automatic Speech Segmentation*. International Journal of Computer Science and Mobile Computing, vol. 6, no. 4, pages 308–315, 2017. (Cited on page 147.)

[Sapp 2007] Craig Stuart Sapp. *Comparative Analysis of Multiple Musical Performances*. In 8th International Society for Music Information Retrieval Conference, pages 497–500, 2007. (Cited on page 127.)

[Sapp 2011] Craig Stuart Sapp. *Computational Methods for the Analysis of Musical Structure*. PhD thesis, Stanford University, 2011. (Cited on page 127.)

[Sargent 2017] Gabriel Sargent, Frédéric Bimbot and Emmanuel Vincent. *Estimating the Structural Segmentation of Popular Music Pieces Under Regularity Constraints*. IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 2, pages 344–358, 2017. (Cited on page 149.)

[Serra 1982] Jean Serra. Image Analysis and Mathematical Morphology. Academic Press, London, 1982. (Cited on pages 16, 22, 59 and 89.)

[Serra 1988] Jean Serra. Image Analysis and Mathematical Morphology, Part II: Theoretical Advances. Academic Press, London, 1988. (Cited on page 18.)

[Shi 2021] Zhengshan Shi. *Computational Analysis and Modeling of Expressive Timing in Chopin's Mazurkas*. In 22nd International Society for Music Information Retrieval Conference, pages 650–656, 2021. (Cited on page 127.)

[Smith 2001] Lloyd Smith and Richard Medina. *Discovering Themes by Exact Pattern Matching*. In 2nd International Conference on Music Information Retrieval, pages 31–32, 2001. (Cited on page 30.)

[Smith 2014] Jordan B.L. Smith, Isaac Schankler and Elaine Chew. *Listening as a Creative Act: Meaningful Differences in Structural Annotations of Improvised Performances*. Music Theory Online, vol. 20, no. 3, 2014. (Cited on page 148.)

[Smith 2017] Jordan B.L. Smith and Elaine Chew. *Automatic Interpretation of Music Structure Analyses: a Validated Technique for Post-hoc Estimation of the Rationale for an Annotation*. In 18th International Society for Music Information Retrieval Conference, pages 435–441, Suzhou, China, 2017. (Cited on page 148.)

[Stowell 2013] Dan Stowell and Elaine Chew. *Maximum a Posteriori Estimation of Piecewise Arcs in Tempo Time-Series*. In From Sounds to Music and Emotions, pages 387–399, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. (Cited on pages 127, 147 and 149.)

[Tekman 1997] Hasan Gürkan Tekman. *Interactions of Perceived Intensity, Duration, and Pitch in Pure Tone Sequences.* Music Perception, vol. 14, no. 3, pages 281–294, 1997. (Cited on page 129.)

[Temperley 1999] David Temperley. *What's Key for Key? The Krumhansl-Schmuckler Key-finding Algorithm Reconsidered.* Music Perception, vol. 17, no. 1, pages 65–100, 1999. (Cited on page 117.)

[Temperley 2004] David Temperley. The Cognition of Basic Musical Structures. MIT Press, 2004. (Cited on page 104.)

[Teytaut 2022] Yann Teytaut, Baptiste Bouvier and Axel Roebel. *A Study on Constraining Connectionist Temporal Classification for Temporal Audio Alignment.* In Interspeech 2022, pages 5015–5019, 2022. (Cited on page 140.)

[Todd 1992] Neil P. McAngus Todd. *The Dynamics of Dynamics: A Model of Musical Expression.* Journal of the Acoustical Society of America, vol. 91, no. 6, pages 3540–3550, 1992. (Cited on pages 127, 147 and 149.)

[Toussaint 2005] Godfried Toussaint. *The Euclidean Algorithm Generates Traditional Musical Rhythms.* In BRIDGES: Mathematical Connections in Art, Music and Science, pages 47–56, 2005. (Cited on pages 3 and 171.)

[Toussaint 2019] Godfried Toussaint. The Geometry of Musical Rhythm: What Makes a "Good" Rhythm Good? CRC Press, 2019. (Cited on pages 22, 23 and 171.)

[Trehub 1984] Sandra E. Trehub, Dale Bull and Leigh A. Thorpe. *Infants' Perception of Melodies: The Role of Melodic Contour.* Child Development, vol. 55, no. 3, pages 821–830, 1984. (Cited on page 92.)

[Tymoczko 2010] Dmitri Tymoczko. A Geometry of Music: Harmony and Counterpoint in the Extended Common Practice. Oxford University Press, 2010. (Cited on page 3.)

[Ukkonen 2003] Esko Ukkonen, Kjell Lemström and Veli Mäkinen. *Geometric Algorithms for Transposition Invariant Content-Based Music Retrieval.* In 4th International Conference of Music Information Retrieval, page 193–199, 2003. (Cited on page 41.)

[Villani 2009] Cédric Villani. Optimal Transport: Old and New, volume 338. Springer Science & Business Media, 2009. (Cited on page 155.)

[Werman 1985] Michael Werman, Shmuel Peleg and Azriel Rosenfeld. *A Distance Metric for Multidimensional Histograms.* Computer Vision, Graphics, and Image Processing, vol. 32, no. 3, pages 328–336, 1985. (Cited on page 156.)

[Widmer 2003] Gerhard Widmer and Asmir Tobudic. *Playing Mozart by Analogy: Learning Multi-Level Timing and Dynamics Strategies*. Journal of New Music Research, vol. 21, no. 1, pages 259–268, 2003. (Cited on pages 127 and 147.)

[Widmer 2004] Gerhard Widmer and Werner Goebl. *Computational Models of Expressive Music Performance: The State of the Art*. Journal of New Music Research, vol. 33, no. 3, pages 203–216, 2004. (Cited on page 126.)

[Wiggins 2002] Geraint A Wiggins, Kjell Lemström and David Meredith. *SIA(M)ESE: An Algorithm for Transposition Invariant, Polyphonic Content-Based Music Retrieval*. In 3rd International Society for Music Information Retrieval Conference, 2002. (Cited on page 40.)

[Witkowska-Zaremba 2000] Elżbieta Witkowska-Zaremba. *Versification, Syntax and Form in Chopin's Mazurkas*. Polish Music Journal, vol. 3, no. 1, 2000. (Cited on pages 117 and 154.)

[Yust 2018] Jason Yust. Organized Time: Rhythm, Tonality, and Form. Oxford University Press, 2018. (Cited on page 167.)

[Zacks 2007] Jeffrey M. Zacks and Khena M. Swallow. *Event Segmentation*. Current Directions in Psychological Science, vol. 16, no. 2, pages 80–84, 2007. (Cited on page 147.)