Persistent Homology and Topological Data Analysis Applied to Music

Internship supervised by Moreno Andreatta

Guillaume Rousseau
guillaume.rousseau@ens-lyon.fr
http://perso.ens-lyon.fr/guillaume.rousseau/

June 18, 2021

Contents

Foreword Before we begin, I would like to thank the Institute de Recherche Mathématique Avancée (IRMA), and particularly Moreno Andreatta, for taking me in for an internship on such a topic and during the covid-19 crisis. As a result of the covid situation, the internship was conducted entirely through telework. I would also like to thank Pierre Guillot and Victoria Callet, who were also part of the team, for the huge help and insight that they offered me.

1. Introduction

This report presents the sum of my work during my 2021 5 months internship, at IRMA. I was part of a small team which is part of a bigger project hosted by the University of Strasbourg, in collaboration with IRCAM (Institut de Recherche et Coordination Acoustique/Musique), called SMIR (Structural Music Information Research). The goal of this project is to explore various ways of representing music in formal mathematical frameworks, in order to analyze music, find patterns in it, and even create new ways of thinking about music.

Links between music and mathematics have been studied for centuries. Music in itself is intrically mathematical in nature, but as a form of art it is also infinitely complex. As such, in order to fully study and understand the links between music and mathematics, it is necessary to use computer science as an interface.

The branch of the SMIR project on which I worked studies music as a topological object. Mattia Bergomi's thesis [1] is a good recount of the many different ways of using topology to study musical objects. The precise subject on which I worked is *persistent homology*, a generalization of homology, which is a well-studied field of algebraic topology. In both persistent and regular homology theory, objects of study are topological spaces. However, in musical and computer science-heavy context, we focus on a specific kind of topological object called *simplicial complexes*, which are finite. In these complexes, computing homology and persistent homology reduces to a computation of a pseudo-Jordan form of some matrices. However, the particular structure of these matrices allows for efficient computation, as described in Zomorodian and Carlsson's article [2]. One of my tasks was implementing an algorithm for the computation of persistent homology in Sage, a programming language used by the team at IRMA.

The current questions which the team working at IRMA are trying to answer are thus the following: how we may turn a piece of music, a partition or MIDI file, into a simplicial complex, and how we may interpret the topological fingerprint of such a complex, musically speaking. In this report, I present some methods which the team and I studied and implemented in order to answer the previous questions.

Report Structure This report has a heavy emphasis on presenting previous works and explaining how they fit together. I first give a short overview of some interesting apparitions of mathematics in musical theory. I then present the Tonnetz, a centuries old mathematical and musical concept which represents harmony and musical paths, and which is the most striking example of simplicial complexes used to formalize musical structures. This acts as a transition to homology, a branch of algebraic topology which studies holes in topological shapes. After presenting homology theory, I expand the notion to persistent homology, which allows for an even more precise study of shape, and which is particularly used in image recognition. I give a fundamental example called the Rips-Vietoris complex, a way of studying the persistent homology of a cloud of points. Then I show some of the techniques which we used to construct simplicial complexes from musical data.

Sections 2, 3, 4 are mostly bibliography and presentations of already established concepts, section 5 contains some implementations of well-known concepts, and section 6 describes the progress of my research on the open problems studied by the team in which I worked. Appendix A, which did not fit in the main body of the report, describes an orthogonal approach to musical analysis through algebraic methods, namely through Fourier analysis. In this section I discuss some of my implementations and experiments on concepts first introduced by Emmanuel Amiot [3], with whom I exchanged during my internship. The second annex contains some additional figures.

2. Preliminaries

In this section I recall some basic notions of musical theory. I also exhibit some examples of mathematics that arise from these notions.

2.1. Notes and Frequences

A music note is simply the frequency of a sound wave. You may have heard the expression "A (La) 440Hz" before: this refers to the note that results from a sound wave with fundamental frequency 440Hz. This note is often used as a reference to calibrate (or tune) instruments. In Western music, there are twelve different notes, which repeat over and over. As said before, the frequency 440Hz corresponds to the note A. But the frequencies 220Hz and 880Hz are also called A. More generally, by doubling the frequency of a sound, its note remains the same. Since there are twelve notes in the equal tempered system, the ratio between two consecutive notes is $\sqrt[12]{2}$. Note that historically, some cultures have divided the octave (the interval between a note and the note with twice the frequency) differently. However since 12-note theory is much more common, I exclusively use it in this report. See Fig. 1 for a table of the powers of $\sqrt[12]{2}$, along with some (rough) fractional approximations.

i	interval name	$\sqrt[12]{2}^i$
0		1
1	Half-tone	1.059
2	Tone	1.122
3	Minor Third	$1.189 \approx \frac{6}{5}$
4	Major Third	$1.259 \approx \frac{5}{4}$
5	Fourth	$1.335 \approx \frac{4}{3}$
6	Tritone	1.412
7	Fifth	$1.498 \approx \frac{3}{2}$
8	Minor Sixth	$1.587 \approx \frac{8}{5}$
9	Major Sixth	$1.682 \approx \frac{5}{3}$
10	Minor Seventh	1.782
11	Major Seventh	1.888
12	Octave	2

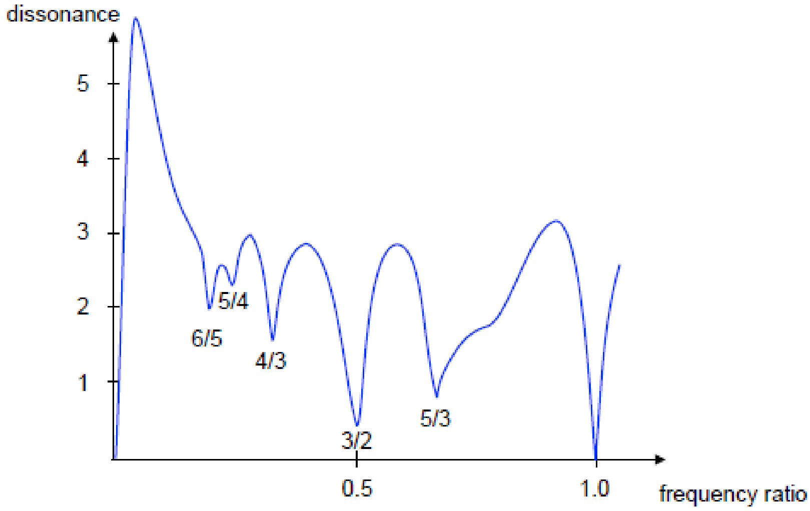


Figure 1: Intervals, and dissonance as a function of frequency ratio

An observation which was made by music theorists is that frequency ratios which are close to small fractions are most pleasing to the ear. On the contrary, $\sqrt[12]{2}^6 = \sqrt{2}$, which is not easily approximated by fractions, is considered to be extremely dissonant. In fact, although most instruments today use exact notes, spaced evenly as above, until the Middle Ages, instruments were tuned using the pythagorean tuning, which is characterized by the fifth (the interval corresponding to 7 semi-tones) being *exactly* $\frac{81}{64}$ and not $\sqrt[12]{2}^7$. Cognitive studies have been conducted on the subject of perception of music. In the 1960's, Plomp and Levelt studied the subject, and plotted perceived dissonance as a function of frequency ratio between two notes (see Fig. 1).

Since talking in terms of ratios and frequencies is rather cumbersome, we consider notes as elements of $\mathbb{Z}$. An octave corresponds to an interval of 12, and two consecutive notes differ by 1. Often, we do not care about the height of a note, but only about its value modulo 12. In this case, we work in $\mathbb{Z}_{12}$ seen as an additive group, ususally identifying 0 with the note C. This group can be generated by 4 elements: 1, 5, 7 and 11. Musically speaking, this means for instance that by playing a note, then the note 7 steps higher, then the next, and so on, it is possible to cover all 12 notes (this is called the cycle of fifths). However if one were to play every fourth note, it would result in a 3-cycle: 0,4,8,0,4,8,...

2.2. Chords

When studying music, we do not only care about singular notes but about sets of notes being played together, *i.e.* chords. A common way of representing notes and chords is to evenly space all 12 notes on the circle, and represent a chord as the convex hull of all notes played (Fig. 2). Among the most basic chords are major chords, which are of the form $\{a, a+4, a+7\}$, and minor chords, of the form $\{a, a+3, a+7\}$. On Fig. 2, a major chord along with three minor chords are represented on the circle. Notice that each minor chord has two notes in common with the major chord, and that each is obtained by applying a symmetry along some axis to the major chord. Using this geometrical representation, we can see how the diedral group D_{12} acts on the set of major and minor chords:

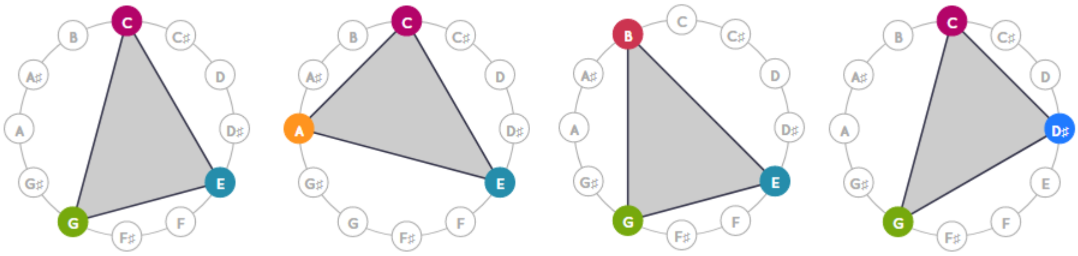


Figure 2: Four chords: C major, A minor, E minor, and C minor

3. The Tonnetz Space

3.1. Neo-Riemannian Tonnetz

As we saw at the end of §2, each major chord can be turned into 3 minor chords by 3 operations which correspond to symetries on the geometric representation of chords:

- The operation which turns C major into C minor (and vice-versa) is called P (for parallel).
- The operation which turns C major into A minor (and vice-versa) is called R (for relative).
- The operation which turns C major into E minor (and vice-versa) is called L (for leading-tone).

Each operation acts similarly by translation on all major and minor triads.

Moreover, we have $R(LR)^3 = P$. These three operations are central in neo-Riemannian musical analysis. The space of triads (major and minor chords) can be represented by an object called the Tonnetz (Fig.3), designed by Euler. As originally shown by Louis Bigo in [4] and successively discussed in [5], The Tonnetz is a type of object called a simplicial complex: it contains vertices (the 12 notes), edges (2-note chords) and triangles (triads). There are three directional axes in the Tonnetz. Each axis correponds to one of the P, L, R operations: taking the symmetric of a triangle w.r.t an axis yields the triangle which is the result of the corresponding operation. Moreover, moving along an axis from a note results in adding $\pm 3, 4, 5$ to the note, depending on the axis and direction chosen. Notice that this corresponds to the 6 most consonant intervals.

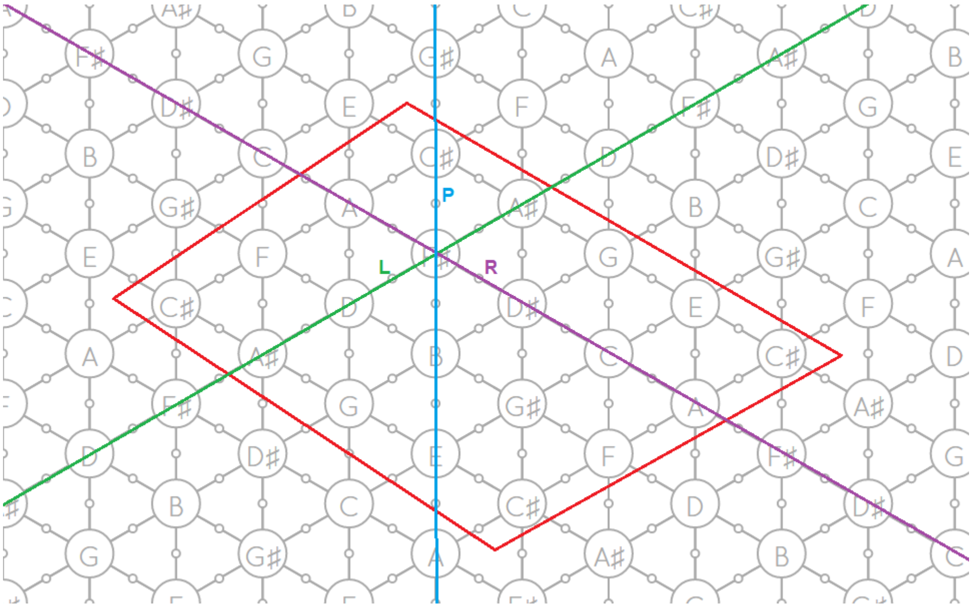


Figure 3: The Tonnetz $T[3,4,5]$ and the P, L, R axes.
The paving pattern is outlined in red.

In Fig.3, the Tonnetz is represented as an infinite paving of the plane. But it is a finite object. Notice that the part outlined in red in the figure has all four corners equal to $C\#$. Moreover, the four edges are pairwise identical. A reader acquainted with topology will have recognized that the Tonnetz is a torus ! We will see more topological features of the Tonnetz later.

3.2. More Tonnetze

Notice that the Tonnetz space is characterized by the fact that moving along axes adds or removes 3,4 or 5 to a note, which is why it is sometimes denoted $T[3,4,5]$. In fact, only two of those three numbers are required, since the third is deduced from the fact that $3+4+5=12$, in other words by taking each axis once, you go around a triangle and end back at your starting point.

In fact, we can consider other Tonnetze, where axes correspond to different intervals. For instance, $T[2,3,7]$ is obtained when the three axes correspond to intervals of 2, 3, and 7, rather than 3, 4 and 5. In this Tonnetz, the chords are no longer the minor and major triads, but some other types of chords. Again, only two of the three numbers suffice since the sum must be equal to 12. Topologically speaking, $T[2,3,7]$ is also a torus. More generally, given $a,b,c \in \mathbb{Z}_{12}$ such that $a+b+c=12$, we can consider the Tonnetz $T[a,b,c]$. Each Tonnetz has its own topological and algebraic features. In §4.4, we study some of those features.

Finally, the Tonnetz only encompass three-note chords, but a reader familiar with modern music theory may know that seventh chords, which are triads with an extra note, are widely used in music. In [6, 7], authors study generalized Tonnetz, in which seventh chords and other four-note chords may be considered.

4. Homology

In this section, I recall some notions of algebraic topology. I begin by giving an informal presentation of the subject, to introduce the notions, and then give the precise mathematical formulation. Finally, we look at the Tonnetz, described earlier, as a topological object and study its homology.

4.1. General Homology

Studying the homology of a topological space is essentially studying its holes. I do not define precisely what a hole is for now, as it is quite complicated in general, and we only study holes in a specific context in the following sections of the report.

Before talking about holes, we must define boundaries. The boundary of an n-dimensional set is the $(n-1)$-dimensional set which forms the outside of the set. It is easy to understand intuitively: the boundary of a segment, which is a 1D set, is its two extremities, the boundary of a disc, which is a 2D set, is the outer circle, the boundary of a ball is the outer sphere, and so on...

Informally, a hole of dimension n in a topological space is a set of dimension $n-1$ which has no boundary, and which is not the boundary of any part of the space. One may also think of an n-hole as an n-dimensional set which is not in the space, enclosed by an $n-1$-dimensional set which is in the space.

For instance, consider the subset $S \subseteq \mathbb{R}^2$ defined in Fig.4. Take the inner circle. It has no boundary, since it is a segment closed on itself, and it is not itself a boundary since the inner disc is not part of the space.

We call Betti number of dimension n the number of n-dimensional holes in a space and denote it β_n. In the previous example, the donut space has $\beta_1 = 1$.

Now consider the torus, with an empty interior. It is connected, so $\beta_0 = 1$. It has one hole in the center, and one hole around the rim, so $\beta_1 = 2$. Additionally, it has a 2D hole: the whole interior of the torus. So, $\beta_2 = 2$. However, one could argue that the torus has infinitely many 1D holes: any circle going around the rim is a hole. This is why we consider *algebraic* topology. In homology theory, we identify two holes when it is possible to continuously deform one into the other. I do not go into any more details on this as it gets mathematically very heavy, and is of no interest in this report.

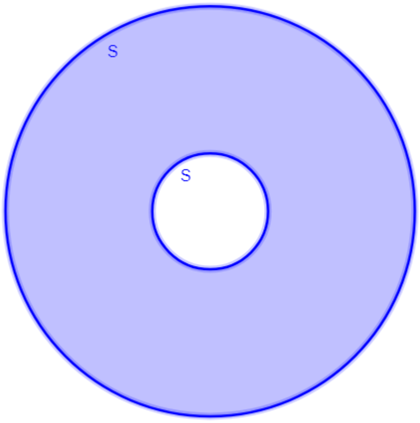


Figure 4: A topological space with one 1D hole

4.2. Simplicial Complexes

To make things easier, we consider very specific kinds of topological space, called simplicial complexes. A simplicial complexes is made up of simplices, which are defined as follows:

Definition 1. *An n-dimensional simplex, or n-simplex, is a set of $n+1$ points $S = \{v_0, ...v_{n+1}\}$, called its vertices. An n-simplex has $n+1$ faces, each being a simplex of dimension $n-1$.*

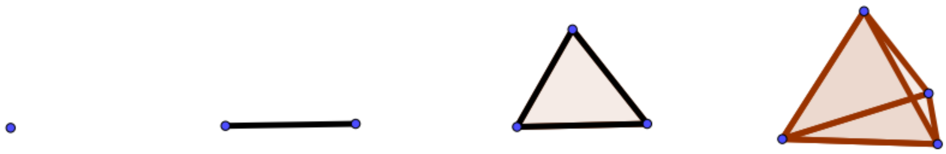


Figure 5: Simplices of dimension 0, 1, 2, 3

Example 1. *For instance, 0-simplices are vertices, 1-simplices are edges, 2-simplices triangles, and so on. (Fig. 5)*

Topologically, an n-simplex can be thought of as the convex hull of its points, living in an n-dimensional space. By glueing simplices together, we obtain a simplicial complex:

Definition 2. *A simplicial complex is a set of simplices $K = \{S_1, \ldots, S_k\}$, such that for all simplex $S \in K$, all faces of S are also in K.*

Notice that there is a difference between the two following complexes:

$$K_1 = \{[0],[1],[2],[0,1],[1,2],[2,0]\}$$
$$K_2 = \{[0],[1],[2],[0,1],[1,2],[2,0],[0,1,2]\}$$

Indeed, topologically K_2 is a full triangle whereas K_1 is a 3-cycle, with no interior.

4.3. Simplicial Homology

We can now formally explain what homology is. First, let us see a few definitions.

Definition 3. *Let $K = \{S_1, ...S_k\}$ be a simplicial complex. Let $\mathbb{K}$ be a field. (typically, $\mathbb{Z}/p\mathbb{Z}$ with p prime)*

For each dimension d, we define the d-th chain group of K to be the free $\mathbb{K}$ module over the d-simplices of K. We denote this module by $\mathbf{C}_d(K)$. An element of $\mathbf{C}_d(K)$ is called a simplicial chain of dimension d, or simply a d-chain.

Let us see what this means, in the simplest case: $\mathbb{K} = \{0, 1\}$. In this context, a chain is a subset of K comprising only d-simplices, and addition is symmetric difference. Here, chain groups are indeed groups because multiplication by 0 and 1 are trivial operations, but in general chain groups are modules, with a full-fledged external multiplication law, and thus the term *chain group* is an abuse of language.

By themselves, chain groups are not very interesting: they do not contain any information besides which simplices are in the complex. However, chain groups are related by functions called the boundary maps.

Definition 4. *The d-th boundary map $\delta_d : C_d(K) \mapsto C_{d-1}(K)$ is defined on simplices as:*

$$\delta_d([s_0, ..., s_d]) = \sum_{i=0}^{d} (-1)^i [s_0, ...\hat{s}_i, ...s_d]$$

In english: the boundary of a simplex is the (formal) alternate sum of its faces.

Notice that when working modulo 2, the boundary is simply the sum of the faces. This is very convenient for notations, and so in the rest of the report we assume that we are working in $\mathbb{Z}_2$, but everything described works in the general case.

We define two subgroups of $\mathbf{C}_d(K)$:

Definition 5. *Let K be a simplicial complex and $\mathbf{C}_d$ its d-th chain group.*

- *We call $Z_d = \ker(\delta_d) \subseteq \mathbf{C}_d$ the group of d-cycles.*
- *We call $B_d = \mathrm{Im}(\delta_{d+1}) \subseteq \mathbf{C}_d$ the boundary group.*

It is easy to check that the boundary maps satisfy $\delta_d \circ \delta_{d+1} = 0$. As a result, we have $Z_d \subseteq B_d$. Thus we can define their quotient.

Definition 6. *The d-th homology group $H_d(K)$ is Z_d/B_d. The d-th Betti number of K is the dimension of $H_d(K)$.*

Informally, the homology elements are cycles, i.e. closed d-surfaces, such that one can "move along" $d+1$ surfaces freely. This corresponds to the notion of hole described at the beginning of this section. The Betti numbers count the number of holes.

4.4. Homology of the Tonnetz

Let us see an example with the $T[3,4,5]$ Tonnetz. As we saw in §3, the Tonnetz is a torus. As such, its first three Betti numbers are:

$$\beta_0 = 1 \quad \beta_1 = 2 \quad \beta_2 = 1$$

Indeed, a torus has one connected component, two 1D hole, and one 2D hole (also called a void). Recall that there are 12 different Tonnetze. Paul Lascabettes, who did an internship in the same team of IRMA in 2019 [8], computed the betti numbers for all Tonnetez. (see Fig.6).

Tonnetz	Nombres de Betti		
	β_0	β_1	β_2
$T[1,2,9]$	1	2	1
$T[1,3,8]$	1	2	1
$T[1,4,7]$	1	2	1
$T[2,3,7]$	1	2	1
$T[3,4,5]$	1	2	1
$T[1,1,10]$	1	1	0
$T[2,5,5]$	1	1	0
$T[2,2,8]$	2	2	0
$T[1,5,6]$	1	1	6
$T[2,4,6]$	2	2	6
$T[3,3,6]$	3	0	3
$T[4,4,4]$	4	0	0

Figure 6: Betti numbers of the Tonnetze

5. Topological Data Analysis and Persistent Homology

5.1. Filtered Simplicial Complexes

Homology is the study of topological features such as the number and agency of holes. It is a very convenient theoretical tool, but it does not apply well to real life data analysis. On many occasions, real life data is modelized not by a simplicial complex, but as a sequence of inclusion-wise increasing simplicial complexes, which approximates the data.

For instance, suppose we have a set of points which approximates a donut, and that we wish to recover some features of the donut, let's say its homology. We can modelize this set of points by an infinite sequence of simplicial complexes $(K_r)_{r\in\mathbb{R}^+}$, indexed by $\mathbb{R}^+$, where K_r contains all simplices $v_0, ...v_d$ such that the v_i's are pairwise less than r units away (see Fig. 7). The main topological feature of the donut, i.e. the hole in the middle, appears for a low value of r and remains until very high values of r, where essentially the whole complex is one big clique.

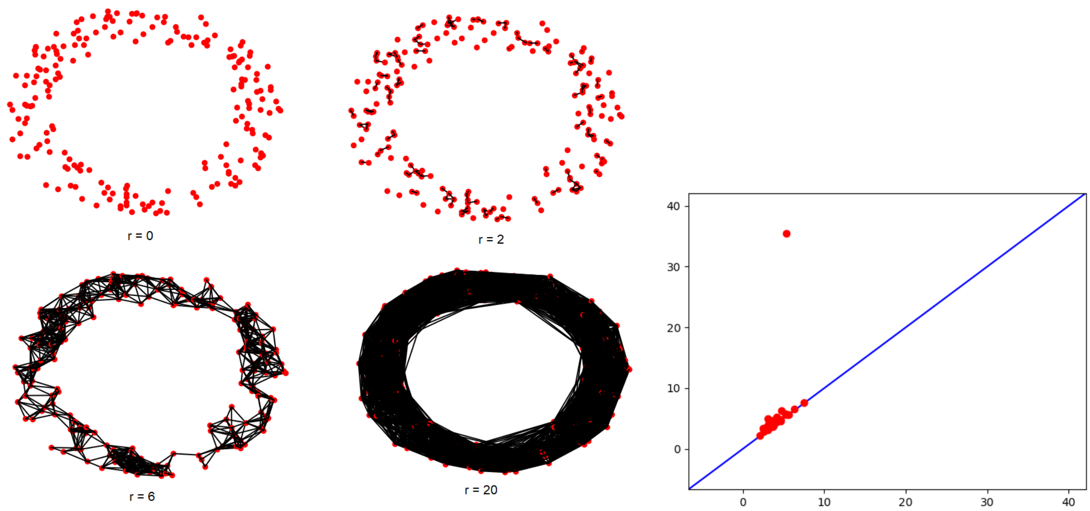


Figure 7: Rips complex K_r for several values of r, and persistence diagram.

Persistent homology is the principal tool of Topological Data Analysis, and is precisely the study of how homological features appear and disappear in such sequences of simplicial complexes. In the donut example, many other holes appear and disappear, but only the hole in the middle remains active for a long time: it has a high *persistence*.

Let us now see formally how persistent homology is defined. First, let us see a proper definition of those sequences of simplicial complexes which are the object of study of this field.

Definition 7. *A* filtered simplicial complex *is a simplicial complex K along with a filtration function $f : K \mapsto \mathbb{R}$ such that for any simplex $\sigma \in K$, for any face τ of σ, $f(\tau) \leq f(\sigma)$. The real value $f(\sigma)$ is called the filtration value of σ.*

For example, the construction given at the beginning of this section (Fig. 7) is a filtered complex: the filtration value of all vertices is 0, the filtration value of an edge is its length, and the filtration value of a simplex is the maximum value of all its faces. This is called a Rips-Vietoris complex [9].

5.2. Persistent Homology

For a given $t \in \mathbb{R}$, we may consider the simplicial complex:

$$K_{\leq t} = \{\sigma \in K | f(\sigma) \leq t\}$$

which contains all simplices of filtration value less than t. Filtered complexes can therefore be seen as a growing family of simplicial complexes indexed by $\mathbb{R}$. The filtration value of a simplex is therefore the index at which the simplex first appears. This is why the filtration value of a simplex is also sometimes called its *degree* in literature. Since there are only a finite number of simplices in a filtered complex, only a finite amount of values t are needed to fully determine the family $(K_{\leq t})$, and as a consequence it can be more intuitive for some to think of filtered complexes as sequences of complexes. Then, given $u \leq v$, consider the homology groups in dimension k, H_k^u and H_k^v. Since $K_{\leq u} \subseteq K_{\leq v}$, we have an inclusion $i : H_k^u \to H_k^v$. The k-th persistent homology module $H_k^{u,v}$ is defined as the image of i.

So, for a single filtered complex, we have an infinity of persistent homology modules, for each dimension. In fact, persistent homology may be finitely described, by a list of *intervals*. Each interval (u, v) corresponds to an element which appears in H_k^u and disappears in H_k^v. in [2], Zomorodian and Carlsson give a formal description of persistent homology, using graded modules, along with an algorithm to compute it.

Example 2. *Let (K, f) be the filtered complex with simplices $0, 1, 01$, with:*

$$\begin{aligned} f(0) &= 0 \\ f(1) &= 1 \\ f(01) &= 3 \end{aligned}$$

Then its 0-intervals are $(0, +\infty)$ and $(1, 3)$. Indeed, until threshold 1, the complex has one connected component, then from 1 until 3 it has two. Then the two are merged into one when the simplex $[0, 1]$ appears, and the remaining connected component lasts infinitely.

5.3. Computing Persistent Homology

One of my main tasks during the internship was implementing an algorithm for the computation of persistent homology in SageMath (`https://www.sagemath.org`), a large-scale math project which adds a whole array of tools from algebra, topology, game theory, combinatorics, and so on, to pythonic language. Sage is widely used across the world, and anyone may participate and submit new features, which is what i did.

```
sage: a = Simplex([0])
sage: b = Simplex([1])
sage: c = Simplex([2])
sage: ab = Simplex([0,1])
sage: X = SimplicialComplex([a,b,c,ab])
sage: X
Simplicial complex with vertex set (0, 1, 2) and facets {(2,), (0, 1)}
sage: X.betti()
{0: 2, 1: 0}
```

Figure 8: Some Sage code

The algorithm which I implemented comes from A. Zomorodian and G. Carlsson [2]. I first wrote a stand-alone version, available at http://www.github.com/quenouillaume/PersiL. This python module implements Zomorodian and Carlsson's algorithm, as well as some tools for the graphical representation of persistent homology (persistence diagrams and barcodes), and for the construction of Rips-Vietoris complexes. I then had to adapt this code to Sage. Regular simplicial homology was already present in Sage (Fig. 8), which meant that a lot of tools needed for persistent homology were already coded. For instance I had coded my own versions of simplices and free modules, which worked but were unoptimized. I was able to replace (and simplify) several parts of my code, and submitted the result to Sage. At the time of submitting this report, my code has not yet been accepted for the next release of Sage, but the process is ongoing. Progress and source code may be found at https://trac.sagemath.org/ticket/31861, on Sage's ticket system site. Here is an example of Sage code that uses my code to compute the persistent homology of Ex.2:

```
sage: a = Simplex([0])
sage: b = Simplex([1])
sage: ab = Simplex([0,1])
sage: X = FilteredSimplicialComplex([(a, 0), (b, 1), (ab, 3)])
sage: X.persistence_intervals(0)
[(1, 3), (0, +Infinity)]
```

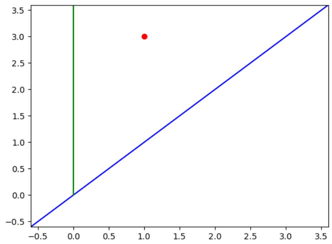
Figure 9: Persistent Homology in Sage

5.4. Analyzing Persistence Diagrams

Persistent homology may be visualized nicely through persistence diagrams. A persistent diagram represents the homology elements, *i.e.* the intervals, on the plane as follows:

- Intervals of the form $(x_0, +\infty)$ are represented as infinite vertical lines of equation $x = x_0$
- Intervals of the form (x_0, x_1) are represented as points (x_0, x_1).

For example, the persistence diagram in dimension 0 of Ex.2 is:



On such a diagram, elements furthest away from the diagonal are the most significant ones, as they last the longest. For instance, the persistence diagram of the Rips complex in Fig.7 has many points near the diagonal, which we may consider as noise, and one point very far from the diagonal which corresponds to the large central hole in the complex.

6. Topological Analysis of Music

One of the goals of the math-music team working in Strasbourg is to create (filtered) simplicial complexes from music, for instance from MIDI files, compute their homology, and use their homological features as fingerprints, in order to recognize or classify music.

In this section, I show some of the ideas that I implemented during my internship to create simplicial complexes from music. Many of those ideas use Vietoris-Rips complexes, and so I implemented Afra Zomorodian's algorithm [10] for a fast construction of VR complexes. I included this code in the PersiL python module.

We modelize a musical piece as a set P of notes.

Definition 8. *A note is a triplet* (b, d, n) *where:*

- b *is the time at which the note begins*
- d *is the duration of the note*
- n *is the pitch of the note*

The temporal distance between two notes (b_1, d_1, n_1) *and* (b_2, d_2, n_2) *is defined as the smallest window of time necessary to hear both notes. Supposing* $b_1 \leq b_2$*, without loss of generality, this distance is:*

$$d((b_1, d_1, n_1), (b_2, d_2, n_2)) = \begin{cases} 0 \ \textit{if}\ b_2 \leq d_1 \\ d_1 - b_2 \ \textit{otherwise} \end{cases}$$

6.1. Pitch-class Rips Complex

Let us start with a toy example. Let P be a musical piece as described above. We build the complex K with vertex set $\mathbb{Z}_{12}$ as follows: The distance between two pitch-classes

i and j is the smallest temporal distance between two notes that are respectively in pitch-classes i and j. From this, we build a Vietoris-Rips complex.

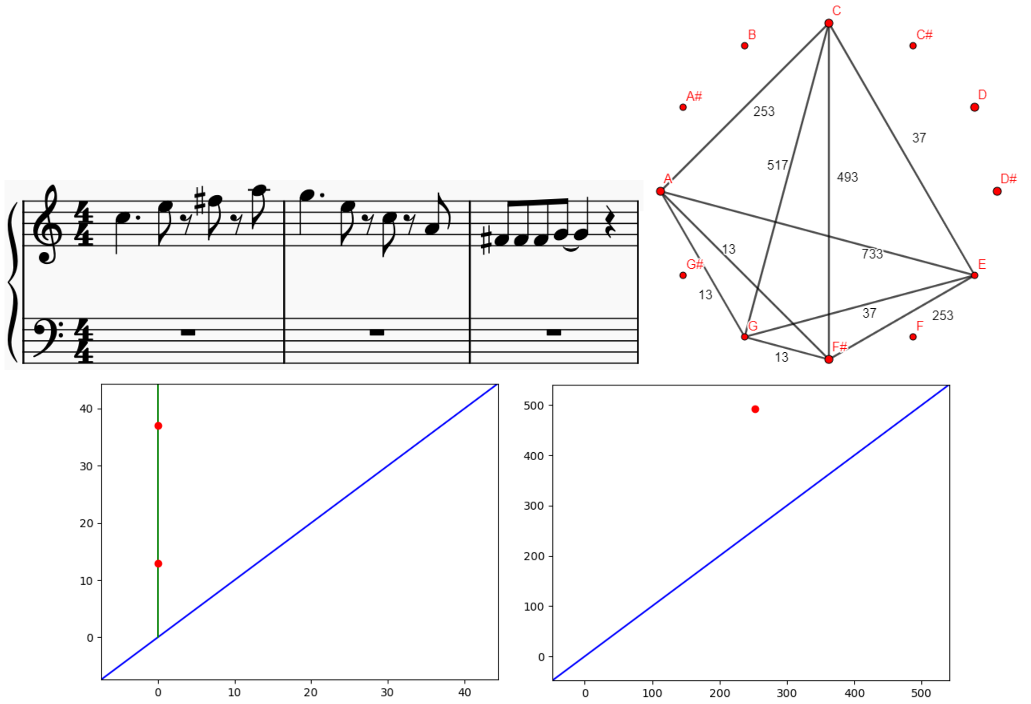


Figure 10: A score, its Rips Complex, and Persistence Diagrams in Dimensions 0 and 1

In Fig. 10, I turned a small score into a MIDI file and then extracted the list of notes as in the format above. I then constructed the Rips-Complex as described above (distances are in an arbitrary MIDI unit). Finally, I computed persistent homology on that complex. Observe that the hole $[C, E] + [E, G] + [G, A] + [A, C]$ appears at time 253. Then, at time 493, the chain $[C, A, F\#] + [A, F\#, G] + [C, E, F\#] + [E, F\#, G]$ appears, and its boundary is exactly the hole. This creates the homology interval $(253, 493)$ visible on the persistence diagram.

The previous examples uses only 5 different notes, and is very short. For larger examples, this method does not yield significant results, as the complexes become increasingly jumbled up. In [1], Mattia Bergomi uses a similar method to construct filtered complexes from music. However, he constructs several complexes from one musical piece: each piece is segmented into small sections, for instance in groups of 1, 2 or 4 bars, and each section is turned into a complex, for which homology is computed. This process yields a sequence of persistence diagrams. Then, further TDA methods such as time series are applied to compare different sequences of diagrams and derive a notion of musical similarity [11].

6.2. Tonnetz-Projected Rips Complex

The team in which I worked during my internship has tried to find ways of turning musical pieces into complexes rather than sequences of complexes. It is clear that this requires a much larger set of vertices. It is also clear that the temporal aspect of music needs to appear somehow in the complex. I will now present one angle of approach that I tried, taking all those things into consideration.

Let P be a set of notes. The vertex set of our complex is now P itself. What happens if we construct a Rips complex directly on the notes of P, with the temporal distance defined above?

Definition 9. *Let P be a list of notes. We define $K_r(P)$ to be the Rips-Vietoris complex constructed from the list P, with temporal distance, with threshold r.*

Since the distance between notes only depends on their beginning and ending times, we have the following:

Lemma 1. *For any $r \geq 0$, the 1-skeleton (i.e. the induced graph) of $K_r(P)$ is an interval graph.*

Proof. Let G be the 1-skeleton of $K_r(P)$. Two notes N_1, N_2 are linked by an edge in G if their temporal distance is less than r. Let $\mathbf{Ext}(b, d, n) = (b - \frac{r}{2}, d + r, n)$ map a note to the same note, starting and ending $\frac{r}{2}$ units later. By definition, (N_1, N_2) is an edge of G if and only if $\mathbf{Ext}(N_1)$ and $\mathbf{Ext}(N_2)$ correspond to overlapping notes. Therefore, G is precisely the interval graph of $\mathbf{Ext}(P)$. □

It is well known that interval graphs do not contain any induced cycles of length 4 or more. As a result, homology in dimension 1 is always trivial, as holes are inevitably closed as soon as they open, and all persistence diagrams obtained this way are empty.

Another issue with the representation above is that it does not account for harmonic features: the distance between two notes is independant of their pitch-classes. In order to fix both problems, I modified the previous idea by factoring in some constraints on notes as follows. Let $T = T[a, b, c]$ be a Tonnetz. Consider the filtered complex $K_r^T(P)$ obtained by projecting $K_r(P)$ onto T, i.e. only keeping edges $((b_1, d_1, n_1), (b_2, d_2, n_2))$ such that (n_1, n_2) is an edge in T. I tested this method with several Tonnetze, on several MIDI files, to see if there was some relevant musical information which appeared on persistence diagrams (see Fig. 11 for an example).

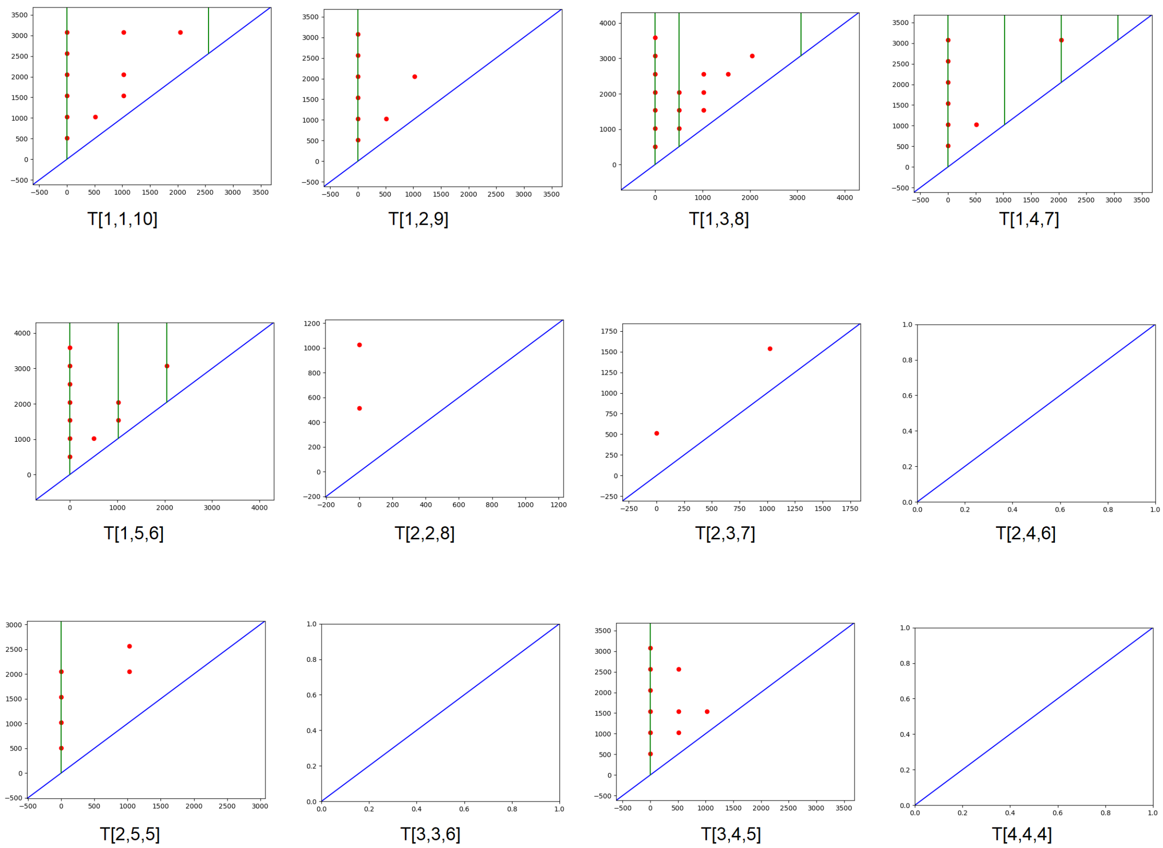


Figure 11: Persistence diagrams for $K_r^T(P)$ for the 1st Bach Chorale

I applied this method on a corpus of MIDI files, including the first 10 Bach chorales, some jazz standards, and some of Debussy's Preludes, to try and see if the similarities and dissimilarities of compositions would appear in the persistence diagrams. Musically, both Bach chorales and the jazz standards used are very tonal, very heptatonic, and use very consonant and usual scales and chords. As such we expect these pieces to work well in the $T[3, 4, 5]$ Tonnetz. However jazz standards also include many chromatisms and tritone chords, and so we expect the $T[1, x, 12 - 1 - x]$ Tonnetz, particularly the $[1, 5, 6]$ one, to be relevant. For Debussy's preludes, their very atypical and modern profiles were expected to yield very different results from the other two datasets.

Unfortunately, it does not seem that the results produced by this method are significant. First, some Tonnetze, such as $T[2, 4, 8]$ and $T[4, 4, 4]$ are not connex, and each connected component is very small. As a consequence, homology derived from them is always trivial. Globally, the persistance diagrams which were produced did not exhibit particular characteristics depending on the musical piece, as we had hoped.

6.3. Rips Complex on Bars

Victoria Callet and Pierre Guillot, who are working in the same team of the SMIR project as Moreno Andreatta and I, have tried a very different approach, on which we have been working together. The idea is to build a Rips complex over the set of bars of a piece. Most musical pieces are divided evenly into bars, which are time-windows. In Fig. 12, two bars are depicted, both with a duration of 4 quarter-notes which is indicate by the time signature at the beginning of the piece. Bars are essentially a mesoscopic scale on which to study music and are usually quite relevant to melody and chord changes.

More precisely, for a set of notes P, we define the $n-th$ bar of length t to be the subset of P containing all notes (b, d, n) such that $[b, b+d] \cap [nt, (n+1)t] \neq \emptyset$. Let us call this bar P_t^n. We then consider the reduced bar $\widehat{P_t^n}$ where the origin of time has been set to nt. In other words, if P_t^n contains a note (b, d, m), then $\widehat{P_t^n}$ contains a note (b', d', m) with:

- $b' = \max(0, b - nt)$
- $d' = \min(t, d' - nt) - b'$

So, for instance, if a note crosses over two bars, it will be in split in two in the reduced bars.

So, we consider the set of all bars of length t, and want to find a proper distance between bars, in order to build a Rips complex.

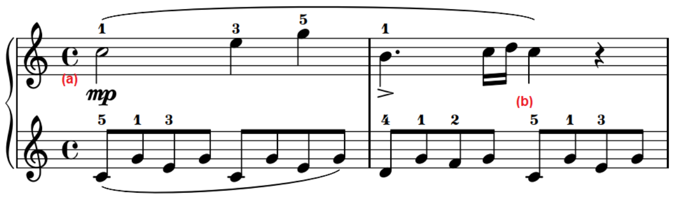


Figure 12: (a) Time signature (b) A bar of length 4 quarter-notes

A simple example of distance between sets is Hausdorff distance.

Definition 10. *Let (X, d) be a metric space and A, B subsets of X. The Hausdorff distance between A and B is defined as:*

$$d_H(A, B) = \max(\sup_{a \in A} d(a, B), \sup_{b \in B} d(b, A))$$

where $d(a, B) = \inf_{b \in B} d(a, b)$

So, we must first define a distance between notes, which will yield, through Hausdorff, a distance between bars. After talking with other members of the team and discussing what they had tried, I tried implementing a distance inheriting from $\|.\|_1$. In other words:

$$d\left((b_1, d_1, n_1), (b_2, d_2, n_2)\right) = |b_1 - b_2| + |d_1 - d_2| + |n_1 - n_2|$$

A simple problem here is that the $|n_1 - n_2|$ component is always trivially small, since MIDI time units are much larger than 100. A solution is to divide the temporal terms by a factor. As of now, the experiments done by the team use a factor such that quarter-notes have value 1. I did not have time to perform many tests using this method, and so cannot present any significant results in my report. I intend to further explore this lead during the time-window between the report deadline and defense date, in order to present some results during the defense.

An idea which was suggested by Mattia Bergomi was to use machine learning to find a good factor: We fix two datasets A and B and want to automatically categorize elements of the two datasets as either A or B. Using machine learning, we test different factors in order to find the best one.

6.4. Visualizing Persistence Diagram Distances

Although the methods above were not fruitful, they still allowed me to develop some ideas on tools to analyze methods and see how relevant they are. Consider a function F which maps musical pieces (or sets of notes) to filtered simplicial complexes. We want to analyze if F is musically relevant, *i.e.* if two musical pieces that are similar produce similar complexes. Let us fix a dimension k, and consider all the k-persistence diagrams from all complexes computed through F. The space of persistence diagrams can be endowed with a number of distances, such as bottleneck distance. To visualize how these persistence diagrams lie in that space, we may want to place them on a plane in a way that preserves distances. Of course, this is rarely possible for questions of dimension. However, many computational models allow for good approximations of this. For instance, in [12], Fruchterman and Reingold describe what is called Force Directed Placement, which is a way of representing undirected weighted graphs in a plane. The idea is that each edge is modelized by a spring of length its weight, after what the position of equilibrium is studied, yielding a placement of nodes which is supposed to approximate the required distances, by minimizing the energy of the system.

Figure 13: FDP of persistence diagrams for some MIDI files

I have applied this visualisation method to some of the methods above, to see if there was not some hidden patterns which we could not directly see on the persistence diagrams themselves. In Fig. 13, some MIDI files from Bach's chorales (BC), jazz standards (JA), the Sacre du Printemps (SA) and other classical pieces (CL) are placed using the FDP method, using complexes constructed in §6.2 with the $T[3,4,5]$ Tonnetz. It seems that on average pieces from a same category are not too far, but there are some outliers, and there could be a bias caused by MIDI files from a same category having been made by the same people (MIDI files made by computer are immediately recognizable from MIDI files made from real human performances). Moreover, because the FDP approach randomizes the initial positions of the nodes before computing position of equilibrium, results between two consecutive runs are inconsistent.

One issue is that Hausdorff distance is *very* sensitive to noise and small variations, which means that the quality of MIDI files has a huge influence on the distances between diagrams. However, it is possible to define a distance between persistence diagrams called *Bottleneck distance*, which has some nice stability properties (see [13] for more):

	BC	CL	JA	SA
BC	4531	4835	4820	4876
CL	4835	4491	4314	4377
JA	4820	4314	4192	4151
SA	4876	4377	4151	4080

Figure 14: Average distances between all 4 categories of the corpus

Definition 11. *Let D_1, D_2 be two persistence diagrams, i.e. multisets of intervals. We consider that D_1, D_2 also each contain all points (x, x) with infinite multiplicity. The bottleneck distance $B(D_1, D_2)$ is defined as:*

$$B(D_1, D_2) = \inf_{\varphi : D_1 \mapsto D_2} \sup_{x \in D_1} \|x - \varphi(x)\|_\infty$$

Where φ denotes a bijection.

The bottleneck distance may be informally understood as a measure of how hard it would be to turn a diagram into the other by moving around points.

Based on a suggestion by Mattia Bergomi, I tried using the bottleneck distance on the persistance diagrams obtained using the different Tonnetz in §6.2. Then, for a pair of MIDI files, I took the maximum distance over all 12 Tonnetz. The results of this seemed promising: on the same corpus as above, graphs produced with the FDP approach were much more consistent. Looking at the numerical values of the distances (Fig. 14), the Bach chorales are on average closer to each other than to other pieces, and the Sacre pieces as well. The CL and JA categories however do not satisfy this condition. This is unconclusive, and could be tried on a larger and better categorized corpus for a definitive answer.

7. Conclusion and Future Work

This internship was conducted entirely from home, due to covid restrictions. As a result, it was hard to focus at times, and I did not progress as fast as I had wished. However, it was a deeply interesting experience. I was able to interact with members of the team often, and was invited to follow a course which was supposed to be only for PhD students, which allowed me to exchange with even more researchers from the math/music field. In particular I was able to exchange with Emmanuel Amiot, and the discussion lead to the development of App. A, which is orthogonal to the topological aspect, but yields some meaningful results.

This internship was heavily code-based and bibliography-based, but I was still able to produce some original work, and although we could not find any groundbreaking results, I hope that the tools which I built will be useful in the future, and that the ideas which we explored still gave some insight as to why they did not lead to significant results. Moreover, we realized how important it is to have a clean database of MIDI files. Indeed, I worked on files which I collected from various internet sites, or made myself. This means that there is a huge discrepancy in quality and in meta-data coherence in my dataset, as between the Bach chorales and other files for instance.

The process of writing and submitting code to Sage was also quite interesting. I had never participed in such a large scale project before, and was surprised to discover the inner workings of Sage. The submitting process is entirely automated, doable from git, and triggers automatic syntax-checking, documentation building and so on... Sage contributors could comment on my code as I submitted it, and I was able to progress quite fast, and learned some interesting programming methodology.

Concerning the future of this research subject: members of the team have talked about creating a large database of clean MIDI files for testing purposes. Moreover, it could be useful, when computing persistent homology, to know not only the number of holes but precisely which elements cause the hole. Indeed, for now it is hard to really understand the link between homological properties and musical properties, and it would be interesting to see which elements of a musical piece cause holes. Unfortunately, we do not know of any method which clearly identifies the holes, perhaps because usually persistent homology is not applied to fields where the vertices of the complexes contain significant information.

Finally, the bottleneck distance is very heavy to compute, although there are some approximations. One of the future subjects of research of the team will be to look into relevant distances between persistence diagrams.

References

[1] Mattia Giuseppe Bergomi. *Dynamical and topological tools for (modern) music analysis*. PhD thesis, 2015. Thèse de doctorat dirigée par Andreatta, Moreno et Haus, Goffredo M. Mathématiques Paris 6 2015.

[2] Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete and Computational Geometry*, 33:249–274, 02 2005.

[3] Emmanuel Amiot. *Music Through Fourier Space*. Springer International Publishing, 2016.

[4] Louis Bigo. *Représentations symboliques musicales et calcul spatial*. PhD thesis, 2013. Thèse de doctorat dirigée par Michel, Olivier Informatique Paris Est 2013.

[5] Louis Bigo and Moreno Andreatta. *Topological Structures in Computer-Aided Music Analysis*, pages 57–80. 10 2016.

[6] Marek Žabka. Generalized tonnetz and well-formed gts: A scale theory inspired by the neo-riemannians. In Elaine Chew, Adrian Childs, and Ching-Hua Chuan, editors, *Mathematics and Computation in Music*, pages 286–298, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[7] Sonia Cannas and Moreno Andreatta. A Generalized Dual of the Tonnetz for Seventh Chords: Mathematical, Computational and Compositional Aspects. In *Proceedings of Bridges 2018: Mathematics, Art, Music, Architecture, Education, Culture*, pages 301–308, Stockholm, Sweden, July 2018.

[8] Paul Lascabettes. Homologie persistante appliquée à la reconnaissance de genres musicaux. 2018.

[9] L. Vietoris. Über den höheren zusammenhang kompakter räume und eine klasse von zusammenhangstreuen abbildungen. *Mathematische Annalen*, 97:454–472.

[10] Afra Zomorodian. Technical section: Fast construction of the vietoris-rips complex. *Comput. Graph.*, 34(3):263–271, June 2010.

[11] Mattia Bergomi and Adriano Baratè. Homological persistence in time series: an application to music classification. *Journal of Mathematics and Music*, 14:1–18, 07 2020.

[12] Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

[13] H. Edelsbrunner and J. Harer. Computational topology - an introduction. 2009.

[14] John Clough and Jack Douthett. Maximally even sets. *Journal of Music Theory*, 35(1/2):93–173, 1991.

[15] Emmanuel Amiot. David lewin and maximally even sets. *Journal of Mathematics and Music*, 1(3):157–172, 2007.

A. Discrete Fourier Analysis

In this section, I discuss some purely algebraic methods which are used in music analysis. Most of the work which I describe comes from [3] and uses Discrete Fourier Transform. I implemented in Python some of those methods, in order to visualize some of the musical characteristics which they exhibit. The work featured in this section stemmed from a course given to PhD students of the Padova university, around maths and music, during which Emmanuel Amiot presented some of his research. I wanted to implement some concepts that he presented, and use them to analyze musical data.

We consider the cyclic group of 12 notes $\mathbb{Z}_{12}$. An element of this group is called a pitch-class, and a subset of $\mathbb{Z}_{12}$ is called a pitch-class set (or PC-set). As explained in 2, a C major chord corresponds to the PC-set $\{0, 4, 7\}$.

For a given PC-set A we consider its characteristic function $1_A : \mathbb{Z}_{12} \to \mathbb{C}$. For instance, the C major (also called diatonic scale) $\{0, 2, 4, 5, 7, 9, 11\}$ is mapped to:

$$\{1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1\}$$

A particularity of the diatonic scale is that it is very even. In fact, it is a maximally even set [14, 15]. Maximally even sets (or MES) satisfy some mathematical equations which express that its elements are as evenly spaced as possible. Among the many properties of MES is the following:

Theorem 1. *Maximally even sets of size d in $\mathbb{Z}_n$ are of the form:*

$$\{\lfloor \frac{nk+a}{d} \rfloor | k = 0, 1, \ldots, d-1\}$$

In other words, MES of size d approximate as well as possible a distribution of d perfectly evenly spaced points. Another interesting property is that the complement of an MES is an MES as well. Applied to the diatonic scale, this means that its complement, the pentatonic scale (the black keys on a piano), is also maximally even. Thus, we have two sets, of size 5 and 7, which are extremely recurring in music, and which have this propery of being very close to perfectly even cycles of size 5 and 7. Another property of MES, which is not particularly useful here but too beautiful to omit, is that given an MES, it either contains or is contained by a translation of its complement. For instance, the pentatonic scale exists on white keys only: $\{0, 2, 4, 7, 9\}$ is a transposition of the black keys by 6 semi-tones.

Now, if we take any PC-set, we may wonder how close it is to these two scales, or to other even or almost even sets of notes. This is the motivation behind the use of Fourier analysis.

Definition 12. *Let $f : \mathbb{Z}_n \to \mathbb{C}$. The Fourier transform of f is $\hat{f} : \mathbb{Z}_n \to \mathbb{C}$ defined as:*

$$\hat{f}(k) = \sum_{k \in \mathbb{Z}_n} f(k) e^{-2ik\pi/n}$$

For a PC-set $A \subseteq \mathbb{Z}_n$, the Fourier transform of A is defined as the Fourier transform of its characteristic function:

$$F(A) = \widehat{1_A} = \sum_{k \in A} e^{-2ik\pi/n}$$

The study of the Fourier coefficient of a PC-set gives some information on the musical nature of the set. Each coefficient is linked with a specific chord or scale, and the amplitude of the coefficient tells how close a PC-set is to the corresponding chord /scale. Let us denote a_k the k-th coefficient:

- a_0 is always an integer, and counts the number of notes in a PC-set.
- a_1 determines the chromatic character, it is maximal for the chromatic hexachord $\{0, 1, 2, 3, 4, 5\}$.
- a_2 determines the quartal/tritonic character, and is maximal for the tritone $\{0, 6\}$
- a_3 determines the augmentedness, and is maximal for the augmented chord $\{0, 4, 8\}$

- a_4 determines the octatonic character, and is maximal for diminished chords $\{0,3,6,9\}$ and unions of diminished chords.

- a_5 determines the diatonic character, it is maximal on the diatonic and pentatonic scales.

- a_6 determines the closeness of a PC-set to the whole-tone scale, which is an evenly spaced set of 6 notes.

These terms come from music theory, if you are not familiar with them, just keep in mind that diatonic and pentatonic scales are extremely prominent in most classical and modern popular music, and are usually considered extremely consonant and tonal, whereas the other scales are considered very atonal. Notice that coefficients a_7 through a_{11} are not described. In fact, it is easy to prove that for all k we have $a_k = a_{12-k}$. In particular, $a_5 = a_7$, hence why the pentatonicism and diatonicism are expressed through the same coefficient.

Let us see some examples. I have taken the average value of the amplitudes of the Fourier coefficients of the bars of several musical pieces, and graphed the results in Fig.15.

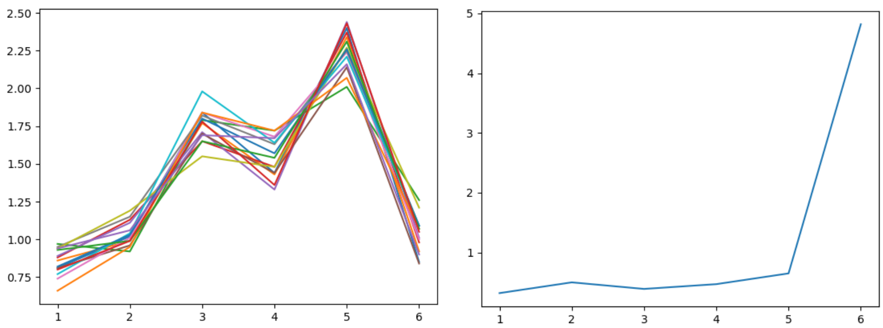


Figure 15: Amplitudes of the fourier coefficients for the first 15 Bach chorales, and Voiles by Debussy.

The first one overlaps the average amplitudes for the first 15 Bach chorales, and the second one shows the same metric for Debussy's Voiles. The difference between the two profiles is striking, and is coherent with the extremely atonal character of Voiles and the extremely tonal character of the chorales. After discussing with E. Amiot, who has greatly developped this field, I wrote some further code to represent the evolution in time of the amplitudes as a musical piece progresses, with a rolling window. This tool allows for a nice visualisation of some musical characteristics. An example can be found at `https://perso.ens-lyon.fr/guillaume.rousseau/02Ichdankdir.gif`, which shows the application of this method to the 2nd Bach chorale.

Fourier analysis is used here for harmonic analysis. However, it may also be used for rhythmic analysis, where subdivisions of the circle in numbers other than 12 appear, and where maximally even sets play key roles as well. For instance, the quintillo and trecillo, which are basic cuban rhythms, are complementary MES of size 5 and 3 respectively, in $\mathbb{Z}_8$.

B. Additional Persistence Diagrams and Figures

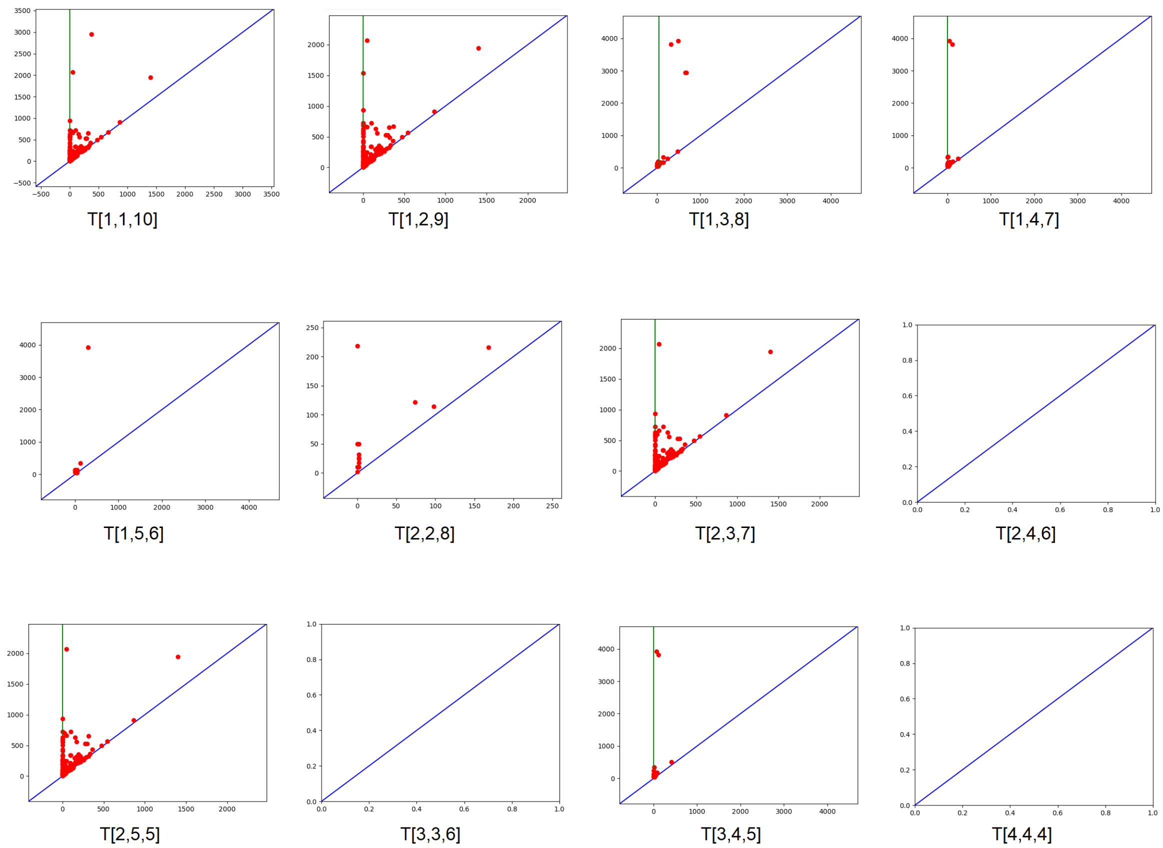


Figure 16: Persistence diagrams for $K_r^T(P)$ for Debussy's Voiles

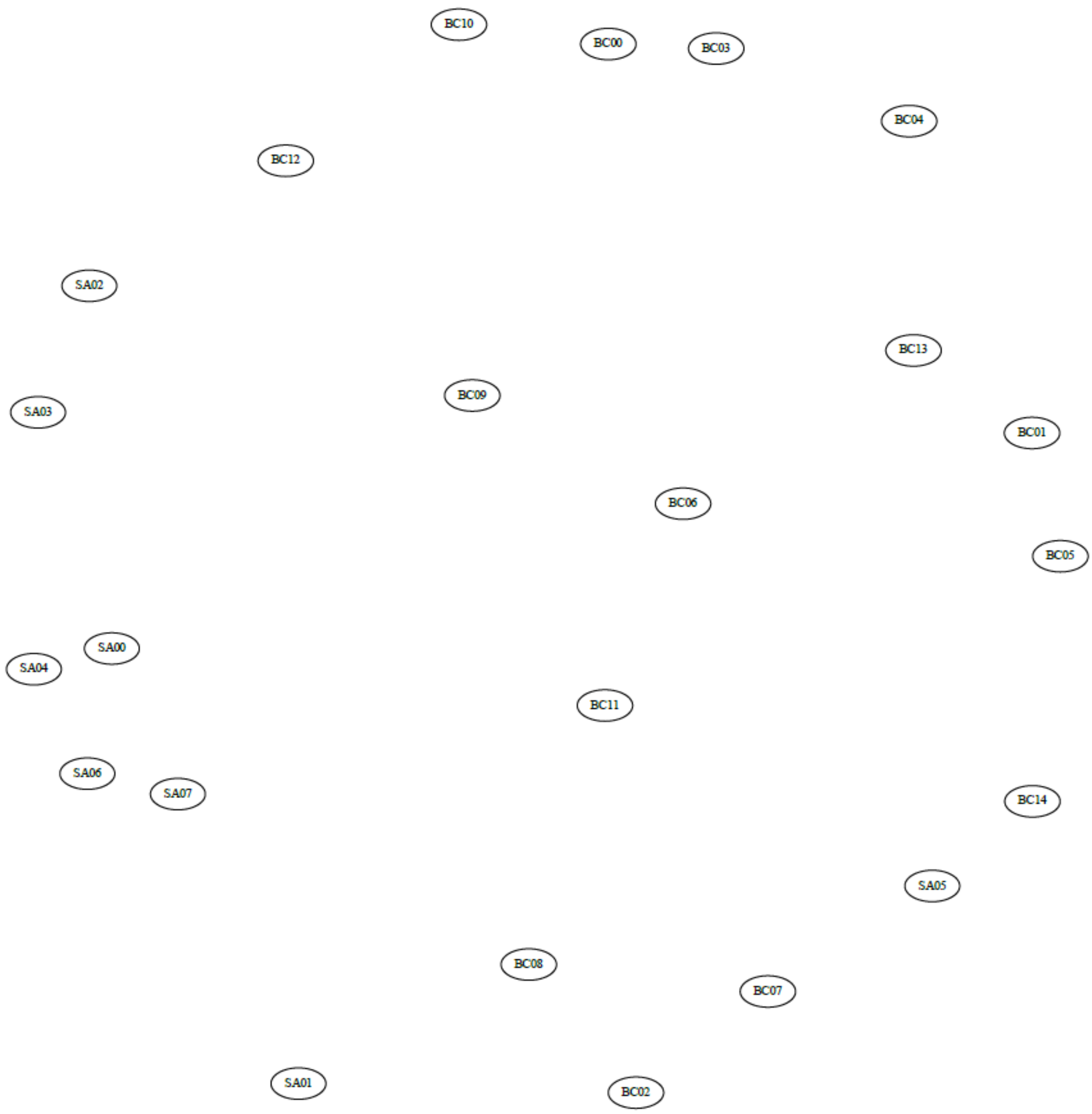


Figure 17: FDP, taking the max bottleneck distance over all 12 Tonnetz-projected persistence diagrams