Structures en combinatoire des mots et en analyse musicale

Florence Levé

▶ **To cite this version:**

Florence Levé. Structures en combinatoire des mots et en analyse musicale. Traitement du texte et du document. Ecole Doctorale en Sciences et Santé de l'Université de Picardie Jules Verne, 2016. tel-01619301

HAL Id: tel-01619301
https://hal.science/tel-01619301

Submitted on 19 Oct 2017

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

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

École Doctorale en Sciences et Santé de l'Université de Picardie Jules Verne

Structures en combinatoire des mots et en analyse musicale

Mémoire d'Habilitation à Diriger les Recherches

Mention Informatique

Florence LEVÉ

Laboratoire MIS
(Modélisation, Information & Systèmes)

Soutenue le mardi 13 décembre 2016 devant le jury constitué de :

Rapporteurs :	M. Jean-Paul Allouche (DR CNRS, IMJ, Univ. Pierre et Marie Curie) M. Frédéric Bimbot (DR CNRS, IRISA Rennes) M. Jean-Frédéric Myoupo (PR, MIS, Univ. de Picardie Jules Verne)
Examinateurs :	M. Marc Chemillier (Directeur d'études à l'EHESS, CAMS) M. El Mustapha Mouaddib (PR, MIS, Univ. de Picardie Jules Verne) M. Patrice Séébold (PR, LIRMM, Univ. de Montpellier) M. Vincent Villain (PR, Garant, MIS, Univ. de Picardie Jules Verne)

Table des matières

Introduction générale **9**

I Contribution à la combinatoire des mots **11**

1 Introduction à la partie **13**

2 Notions préliminaires **15**
2.1 Mots et morphismes 15
2.2 Mots et morphismes sturmiens 17
2.3 Mots et morphismes épisturmiens 18

3 Mots directeurs équivalents pour les mots épisturmiens **21**

4 Quasipériodicité des mots et morphismes sturmiens **25**
4.1 Mots et morphismes quasipériodiques 25
4.2 Mots sturmiens non-quasipériodiques 26
4.3 Morphismes sturmiens quasipériodiques 28

5 Quasipériodicité des mots et morphismes épisturmiens **33**
5.1 Mots épisturmiens quasipériodiques 33
5.1.1 Clôture palindromique dans les mots épisturmiens 33
5.1.2 Quasipériodicité des mots épisturmiens 35
5.2 Quasipériodicité des morphismes épisturmiens 38

6 Conclusion **41**

II Contribution à l'analyse musicale par ordinateur **43**

7 Introduction à la partie **45**

8 Formes musicales : fugues, forme sonate. **47**
8.1 L'analyse de la forme musicale 47
8.2 Analyse automatique de fugues 49
8.2.1 Représentation de la musique 49
8.2.2 Une adaptation de l'algorithme de Mongeau-Sankoff 50

8.2.3 Algorithmes pour l'analyse de fugue . . . 52
8.3 Vers une analyse automatique des formes sonates . . . 59
8.3.1 La forme sonate . . . 59
8.3.2 Analyser informatiquement la forme sonate ? . . . 61
8.3.3 Détection du couple Exposition/Réexposition . . . 63

9 Analyse à partir de fichiers polyphoniques 67
9.1 Extraction du rythme dans la musique polyphonique . . . 67
9.1.1 Représentation de séquences rythmiques monophoniques . . . 68
9.1.2 Extraction de rythme . . . 68
9.1.3 Recherche de similarités rythmiques . . . 69
9.1.4 Discussion . . . 74
9.2 Vers une modélisation de la texture sur des données symboliques . . . 74
9.2.1 Texture musicale . . . 75
9.2.2 Formalisation de la texture . . . 77
9.2.3 Trouver les couches synchronisées . . . 79
9.2.4 Résultats et discussion . . . 82

10 Perspectives 85

Liste de mes publications 89

Bibliographie 91

À la mémoire de mon père

À Gaël, Valentine et Aurélien

Remerciements

Ce mémoire présente l'orientation de mes recherches suite à mon doctorat, de mon recrutement en tant que maître de conférences en 2004 à l'Université de Picardie Jules Verne jusqu'en 2016.

Ces recherches n'auraient pu se faire sans les collaborations que j'ai eues au fil de ces années avec des collègues de grande qualité.

Je remercie en tout premier lieu Gwénaël Richomme, qui depuis la fin de mon doctorat jusqu'à son recrutement en tant que Professeur à Montpellier, a guidé mes recherches et m'a permis de développer la rigueur nécessaire à l'étude des sujets plutôt techniques que nous avons abordés ensemble.

Je remercie également mes collègues d'Algomus à Lille et à Amiens, en particulier Mathieu Giraud, Richard Groult et Emmanuel Leguy : c'est ensemble que depuis 2009 nous avons développé la thématique de l'analyse de partitions, effectuant tous des virages plus ou moins radicaux dans nos recherches, et les perspectives passionnantes qui se sont ouvertes à nous vont sans doute nous occuper encore pendant quelques années.

Je remercie Jean-Paul Allouche et Frédéric Bimbot pour l'intérêt qu'ils ont montré pour mes travaux et pour avoir accepté d'être rapporteurs de mon mémoire. Jean-Frédéric Myoupo a suivi mon évolution depuis de nombreuses années en tant qu'étudiante puis au sein de l'équipe, merci également d'avoir accepté d'être rapporteur de ce mémoire.

Je remercie tout particulièrement Patrice Séébold d'avoir accepté de participer à ce jury. Merci de m'avoir fait découvrir la combinatoire des mots, d'avoir guidé mes premières années de recherche et de m'avoir transmis le goût de la recherche.

Merci également à Marc Chemillier d'avoir bien voulu participer au jury. Ses articles ont été parmi les premiers auxquels je me suis intéressée quand j'ai abordé l'informatique musicale, et restent pour moi une référence dans l'utilisation des mathématiques et des mots pour la compréhension de la musique.

En tant que directeur du laboratoire MIS, El Mustapha Mouaddib m'a toujours encouragée et soutenue dans mes démarches, je le remercie d'avoir accepté de faire partie de ce jury.

Merci enfin à Vincent, parrain de mon habilitation, pour son soutien et la confiance qu'il m'a accordée en me permettant de co-encadrer de jeunes chercheurs et de développer des recherches qui n'étaient pas dans ma thématique d'origine.

Je n'oublie pas les autres chercheurs avec qui j'ai eu l'occasion de travailler, les étudiants que j'ai encadrés, et plus particulièrement Cyril, Khaled et Nicolas, Hervé et toutes les personnes qui ont collaboré à l'atelier Pierres Musicales, ainsi que l'ensemble des collègues qui m'ont accompagnée

au sein du département informatique. Un merci spécial pour Valérie et Juliette, les secrétaires du laboratoire MIS, pour leur aide précieuse et toujours efficace.

Je terminerai ces remerciements par les personnes qui me sont les plus chères. Valentine, Aurélien, merci pour les moments de bonheur que vous m'apportez tous les jours. Merci Gaël de m'avoir encouragée à passer cette habilitation et aidée au quotidien pour que je puisse aller au bout, malgré les nuits (très) courtes de ces derniers mois, et de m'avoir donné mes plus belles publications (et qui sont aussi de loin les plus auto-citées) : celles de notre livret de famille...

Introduction générale

L'objectif de ce mémoire est de présenter l'orientation de mes travaux de recherches depuis l'obtention de mon doctorat, en novembre 2003. Mes travaux de thèse, effectués sous la direction de Patrice Séébold, portaient sur la combinatoire des mots, en particulier la complexité des mots finis, les conjugués des morphismes standard, et les points fixes de morphismes. Recrutée comme maître de conférences à l'Université de Picardie Jules Verne en 2004, j'ai continué dans la lignée de ces travaux une collaboration fructueuse avec Gwénaël Richomme, sur le problème de la quasipériodicité [LR2004, LR2007a, LR2007b]. Nous avons également travaillé avec Amy Glen (Univ. Murdoch, Perth, Australie) sur le cas particulier de la quasipériodicité des mots et des morphismes épisturmiens [GLR2008a, GLR2008b, GLR2009]. Lorsque Gwénaël a été recruté Professeur à Montpellier, j'ai diversifié mes thématiques de recherche. Nous venions de fusionner l'équipe Mots avec l'équipe Systèmes Distribués, ce qui m'a amenée à collaborer avec Vincent Villain, Yoann Dieudonné et Franck Petit sur le problème de l'élection de leader dans un système anonyme sans orientation en utilisant des mots de Lyndon pour caractériser la position des nœuds dans le système [DLPV2013]. La collaboration avec Vincent Villain s'est poursuivie sur un autre thème, au travers du co-encadrement de la thèse de Khaled Mohamed, sur la stabilisation instantanée dans le modèle à passage de messages, de 2011 à 2015 [LMV2014, LMV2016]. Parallèlement j'ai commencé à collaborer avec Richard Groult et Mathieu Giraud sur des problèmes d'analyse musicale par informatique. Nous avons ainsi créé le groupe de recherche Algomus entre Amiens et Lille, nous intéressant à plusieurs problèmes d'analyse musicale sur des données symboliques, c'est-à-dire une représentation des partitions sous forme textuelle. Nous avons travaillé sur l'extraction de rythme à partir de données polyphoniques [AGGGLS2011], l'analyse automatique de la fugue [GGL2012a, GGL2012b, GGLL2015], l'analyse de la forme sonate [DGGLL2014], la texture musicale [GLMRT2014] et présenté notre vision de l'analyse de structure et de forme dans un chapitre de livre à l'invitation de David Meredith (Aalborg University) [GGL2016]. Nous avons co-encadré plusieurs étudiants sur ces thématiques, et notamment Nicolas Guiomard-Kagan, qui termine actuellement sa thèse à l'Université de Picardie Jules Verne sur l'analyse des données polyphoniques et la séparation de voix [GGGL2014, GGGL2015, GGGL2016]. Ces dernières années ont donc été plus axées sur ce thème que sur mon domaine d'origine, bien que j'aie à nouveau eu l'occasion de collaborer avec Gwénaël Richomme sur la quasipériodicité des morphismes quelconques en 2013 [LR2013], et avec Amy Glen sur une généralisation des mots trapézoïdaux (dont la fonction de complexité a la forme d'un trapèze) à un alphabet quelconque en 2015 [GL2015].

Il ne m'a pas paru adéquat de parler ici de l'ensemble de ces travaux. J'ai préféré me demander quel était le point commun entre ces thématiques. La majeure partie des problèmes que j'ai étudiés depuis 2004 avait pour objectif l'étude de la *structure*. J'ai donc décidé de me focaliser dans ce mémoire sur les problèmes de structure qui ont jalonné mon parcours et les différents moyens de les traiter.

Le mot *structure* vient du latin *structura*, de *struere* : "construire", "assembler". Se rapportant initialement à la manière dont un édifice est constuit, par extension, elle désigne "la manière dont les parties d'un tout sont arrangées entre elles" (dictionnaire Larousse). En combinatoire des mots, l'étude de la structure peut être simplement de regarder la manière dont les lettres sont positionnées dans un mot, ou l'étude des *facteurs* (suites de lettres juxtaposées) et de leurs répétitions. Le problème auquel nous allons nous intéresser ici est celui de la *quasipériodicité*, c'est-à-dire la structure des mots constitués d'un *motif* répété avec ou sans chevauchement. En musique, la structure

peut désigner l'agencement de différentes parties musicales (similaires ou distinctes) pour former un schéma global, apportant un certain équilibre à une œuvre. Nous allons nous intéresser ici aux différents éléments entrant en jeu dans l'analyse de la structure "horizontale" d'une pièce musicale : répétition de *motifs*, *harmonie*, *rythme*, *texture*... et au moyen de les combiner pour aboutir à une analyse musicale par ordinateur proche d'une analyse musicale humaine. Nous verrons aussi que l'on peut étudier localement la structure "verticale" d'une pièce lorsque celle-ci est *polyphonique*, en regardant l'agencement des différentes *voix* entre elles.

Ce mémoire d'habilitation va donc se présenter en deux parties. La première partie, contribution à la combinatoire des mots, traitera du problème de la quasipériodicité, d'abord dans les mots et morphismes sturmiens, puis dans les mots et morphismes épisturmiens, avant d'évoquer en conclusion le cas des morphismes quelconques. La deuxième partie, contribution à l'analyse musicale par ordinateur, traitera de l'analyse de structure et de forme, d'abord sur les données monophoniques, avec l'analyse de la *fugue* et des propositions pour l'analyse de la *forme sonate*, puis sur les données polyphoniques, présentant une approche originale du rythme et de la texture. Je terminerai par mes perspectives de recherche actuelles pour l'amélioration de l'analyse de la structure musicale.

Première partie

Contribution à la combinatoire des mots

Chapitre 1

Introduction à la partie

Les mots qui sont au coeur de cette première partie sont les *mots sturmiens*, qui sont des mots infinis équilibrés et non ultimement périodiques sur un alphabet binaire, et les *mots épisturmiens*, qui sont une de leurs généralisations sur un alphabet quelconque (les définitions seront données dans le chapitre 2).

Ce qui m'intéresse le plus dans ces mots est que leur structure peut être étudiée sous différents angles. D'une part ces mots, étant simplement constitués de lettres, possèdent des facteurs, qui peuvent se répéter, se chevaucher, n'apparaître qu'au début ou tout au long du mot... D'autre part, ces mots peuvent être construits en appliquant une composition infinie de morphismes sur une lettre. Il est alors pertinent de regarder quelles compositions distinctes peuvent permettre de construire les mêmes mots, ou quels mots ne peuvent être construits que par une unique composition de morphismes, et de trouver un moyen d'inférer des propriétés de structure locale par une bonne connaissance de ces compositions. Enfin ces compositions de morphismes peuvent à nouveau être appliquées sur des mots, et modifier leur structure pour faire apparaître un nouvel agencement de facteurs.

La propriété de structure que je vais présenter ici est la quasipériodicité, un mot étant quasipériodique s'il peut être entièrement couvert par un même facteur appelé quasipériode (les différentes occurrences de cette quasipériode pouvant être adjacentes ou chevauchantes).

Après quelques notions préliminaires dans le chapitre 2, en particulier sur les mots et les morphismes sturmiens et épisturmiens, je présente dans le chapitre 3 un résultat obtenu avec Gwénaël Richomme sur la normalisation de la décomposition morphique des mots épisturmiens. Le chapitre 4 présente les résultats obtenus avec G. Richomme sur la quasipériodicité des mots et morphismes sturmiens, et enfin le chapitre 5 présente les résultats obtenus avec G. Richomme et Amy Glen sur la quasipériodicité des mots et morphismes épisturmiens.

Chapitre 2

Notions préliminaires

2.1 Mots et morphismes

Soit $\mathcal{A}$ un *alphabet* fini. Un *mot* fini sur $\mathcal{A}$ est une suite finie de lettres de $\mathcal{A}$. Le *mot vide* ε correspond à la suite vide. Muni de l'opération de concaténation, l'ensemble $\mathcal{A}^*$ de tous les mots finis sur $\mathcal{A}$ forme un *monoïde libre* d'élément neutre ε et d'ensemble de générateurs $\mathcal{A}$. On note $\mathcal{A}^+ = \mathcal{A}^* \setminus \{\varepsilon\}$ l'ensemble des mots *non vides* sur $\mathcal{A}$.

Soit un mot fini $w = x_1x_2\cdots x_m \in \mathcal{A}^+$ tel que $\forall i$, $x_i \in \mathcal{A}$, la *longueur* de w est $|w| = m$. La longueur du mot vide est 0. Le nombre d'occurrences d'une lettre a de $\mathcal{A}$ dans le mot w est noté $|w|_a$. Si $|w|_a = 0$, alors w est dit *sans* a. Pour tout entier $p \geq 1$, la p-ième puissance de w est le mot w^p obtenu en concaténant p occurrences de w.

Exemple 2.1.1. Pour le mot fini $w = abaab$, $A = \{a, b\}$, $|w| = 5$, $|w|_a = 3$, $|w|_b = 2$.

Un *mot infini* (à droite) $\mathbf{x}$ est une suite indicée par $\mathbb{N}^+$ avec des valeurs dans $\mathcal{A}$, i.e., $\mathbf{x} = x_1x_2x_3\cdots$ tel que pour tout i, $x_i \in \mathcal{A}$. L'ensemble des mots infinis sur $\mathcal{A}$ est noté $\mathcal{A}^\omega$. Soit v un mot fini non vide, on note v^ω le mot infini obtenu en concaténant v avec lui-même infiniment souvent. Un mot infini *ultimement périodique* est un mot qui peut être écrit $uv^\omega = uvvv\cdots$, pour u, $v \in \mathcal{A}^*$, $v \neq \varepsilon$. Si $u = \varepsilon$, alors un tel mot est *périodique*. Un mot infini non ultimement périodique est dit *apériodique*. Afin de faciliter la lecture, dans ce qui suit, les mots infinis sont parfois indiqués en caractères gras pour les distinguer des mots finis.

Exemple 2.1.2. Le mot $(ab)^\omega = abababab\ldots$ est périodique de période ab. Le mot $a(ab)^\omega = aababababab\ldots$ est ultimement périodique.

Soit X un ensemble de mots finis, X^* (resp. X^ω) est l'ensemble de tous les mots finis (resp. infinis) pouvant être obtenus en concaténant des mots de X. Le mot vide ε appartient à X^*.

Un mot fini u est un *facteur* d'un mot fini ou infini w si $w = pus$ pour des mots p, s quelconques (où s est infini si et seulement si w est infini). On parlera alors de l'occurrence de u commençant en position $|p| + 1$ ou du mot p précédant l'occurrence de u (u pouvant bien sûr avoir des occurrences à plusieurs positions dans w). Lorsque $u \neq w$, on dit que u est un facteur *propre* de w. Le mot u est appelé *préfixe* (resp. *suffixe*) de w si $p = \varepsilon$ (resp. $s = \varepsilon$). On utilise la notation $p^{-1}w$ (resp. ws^{-1}) pour représenter l'*effacement* d'un préfixe p (resp. suffixe s) du mot w. Un mot u est un *bord* d'un mot w s'il est à la fois préfixe et suffixe de ce mot.

Exemple 2.1.3. Ainsi, le mot $w = abaab$ possède 11 facteurs non vides : a, b, ab, ba, aa, aba, baa, aab, $abaa$, $baab$, $abaab$. Les mots a, ab, aba, $abaa$ sont préfixes propres de w, et l'effacement du préfixe a donne $a^{-1}w = baab$. Le mot ab est un bord de w.

Un mot infini $\mathbf{x} \in \mathcal{A}^\omega$ est appelé *suffixe* de $\mathbf{z} \in \mathcal{A}^\omega$ s'il existe un mot $w \in \mathcal{A}^*$ tel que $\mathbf{z} = w\mathbf{x}$. C'est-à-dire que $\mathbf{x} = \mathrm{T}^{|w|}(\mathbf{z}) = w^{-1}\mathbf{z}$, où T désigne le *décalage* : $\mathrm{T}((x_n)_{n\geq 1}) = (x_{n+1})_{n\geq 1}$. Notons qu'un préfixe ou un suffixe u d'un mot $w \in \mathcal{A}^\infty$ est dit *propre* si $u \neq w$. Pour des mots finis $w \in \mathcal{A}^*$, le décalage T est circulaire, i.e., si $w = xv$ où $x \in \mathcal{A}$, alors $\mathrm{T}(w) = vx$.

L'*alphabet* d'un mot w, noté $\mathrm{Alph}(w)$ est l'ensemble des lettres apparaissant dans w, et si w est infini, on note $\mathrm{Ult}(w)$ l'ensemble des lettres apparaissant infiniment souvent dans w.

Soit un alphabet $\mathcal{A}$, un *morphisme* f sur $\mathcal{A}$ est une application de $\mathcal{A}^*$ dans $\mathcal{A}^*$ telle que $f(uv) = f(u)f(v)$ pour tous mots u, v sur $\mathcal{A}$. Un morphisme sur $\mathcal{A}$ est entièrement défini par les images des lettres de $\mathcal{A}$.

Tous les morphismes considérés dans ce manuscrit seront non effaçants : l'image de tout mot non vide n'est jamais vide. De plus l'action d'un morphisme f sur $\mathcal{A}^*$ peut être naturellement étendue aux mots infinis : l'image d'un mot infini est donc infinie et obtenue en concaténant infiniment les images des lettres du mot (c'est-à-dire que si $\mathbf{x} = x_1x_2x_3\cdots \in \mathcal{A}^\omega$, alors $f(\mathbf{x}) = f(x_1)f(x_2)f(x_3)\cdots$). Un mot w est appelé *point fixe* d'un morphisme f si $f(w) = w$.

La composition des morphismes est notée par juxtaposition.

Soit un ensemble de morphismes X, on notera X^* (resp. X^ω) l'ensemble de toutes les compositions finies (resp. infinies) de morphismes de X. Lorsqu'un mot w est égal à $\lim\limits_{n\to\infty} f_1f_2\ldots f_n(a)$, $f_i \in X$, on dit que w peut être *décomposé* (infiniment) sur X. Rappelons (voir [59, page 9] par exemple) qu'une suite $(u_n)_{n\geq 0}$ de mots finis sur un alphabet A converge en un mot infini x si tout préfixe de x est préfixe de tous les mots u_n à un nombre fini près. Ce mot x est unique et peut être noté $x = \lim\limits_{n\to\infty} u_n$.

Soit f un morphisme, les *puissances* de f sont définies récursivement par $f^0 = Id$ (le morphisme Identité), $f^i = ff^{i-1}$ pour tout entier $i \geq 1$. Quand pour une lettre a donnée, $f(a) = ax$ avec $x \neq \varepsilon$, le morphisme f est dit *prolongeable* en a. Dans ce cas, pour tout $n \geq 0$, $f^n(a)$ est un préfixe de $f^{n+1}(a)$. Si de plus, pour tout $n \geq 0$, $|f^n(a)| < |f^{n+1}(a)|$, la limite $\lim\limits_{n\to\infty} f^n(a)$ est le mot infini noté $f^\omega(a)$ ayant tous les $f^n(a)$ comme préfixes. Cette limite est également un point fixe de f.

Exemple 2.1.4. Le mot de Fibonacci F est le mot infini point fixe du morphisme $\varphi : a \mapsto ab$, $b \mapsto a$ prolongeable en a.
$\varphi(a) = ab$
$\varphi^2(a) = aba$
$\varphi^3(a) = abaab$
...
Ainsi, $F = abaababaabaababaabab\ldots$

Il se trouve que φ est un morphisme dit sturmien, et F est un mot sturmien qui a fait l'objet de nombreuses études. Les mots sturmiens sont définis sur un alphabet contenant deux lettres distinctes. Le chapitre suivant donne les définitions et propriétés principales des mots et morphismes sturmiens et de l'une de leur généralisation sur un alphabet quelconque : les mots et morphismes épisturmiens.

2.2 Mots et morphismes sturmiens

Depuis les travaux originaux de Morse et Hedlund [71], les mots sturmiens ont été beaucoup étudiés et leurs propriétés remarquables ont des liens avec de nombreux domaines de recherche comme la théorie des nombres, la géométrie, les systèmes dynamiques et la combinatoire des mots. Ces mots infinis, définis sur un alphabet binaire, ont de nombreuses définitions équivalentes et caractérisations (voir [59, chapter 2] par exemple).

En particulier, les mots sturmiens sont des mots infinis équilibrés et non ultimement périodiques :

— Un mot w (fini ou infini) sur l'alphabet $\{a, b\}$ est dit *équilibré* si pour tous facteurs u et v de même longueur, $||u|_a - |v|_a| \leq 1$.

— Un mot infini $\mathbf{w}$ est *ultimement périodique* s'il existe des mots finis u et v tels que $w = uv^\omega$.

De nombreux travaux sur les mots sturmiens utilisent les morphismes sturmiens, c'est-à-dire les morphismes qui préservent les mots sturmiens (i.e. qui transforment tout mot sturmien en mot sturmien). Séébold [98] a prouvé que l'ensemble de ces morphismes est $\{E, L_a, L_b, R_a, R_b\}^*$ où E, L_a, L_b, R_a, R_b sont définis par :

$$E : \begin{cases} a \mapsto b \\ b \mapsto a, \end{cases} \quad L_a : \begin{cases} a \mapsto a \\ b \mapsto ab, \end{cases} \quad L_b : \begin{cases} a \mapsto ba \\ b \mapsto b, \end{cases}$$

$$R_a : \begin{cases} a \mapsto a \\ b \mapsto ba, \end{cases} \quad R_b : \begin{cases} a \mapsto ab \\ b \mapsto b. \end{cases}$$

De nombreuses relations existent entre les mots et les morphimes sturmiens. En particulier, Berthé, Holton et Zamboni ont montré que les mots sturmiens peuvent être *décomposés infiniment* sur $\{L_a, L_b, R_a, R_b\}$:

Théorème 2.2.1. [7] *Tout mot sturmien w sur $\{a, b\}$ admet une représentation unique de la forme*[1]

$$w = \lim_{n\to\infty} L_a^{d_1-c_1} R_a^{c_1} L_b^{d_2-c_2} R_b^{c_2} \dots L_a^{d_{2n-1}-c_{2n-1}} R_a^{c_{2n-1}} L_b^{d_{2n}-c_{2n}} R_b^{c_{2n}}(a)$$

où $d_k \geq c_k \geq 0$ pour tous entiers $k \geq 1$, $d_k \geq 1$ pour $k \geq 2$ et si $c_k = d_k$, alors $c_{k-1} = 0$.

Une famille particulièrement bien connue des mots sturmiens est l'ensemble des *mots sturmiens standard* (ou caractéristiques). Ils correspondent au cas où pour tout $k \geq 0$, $c_k = 0$. Par conséquent tout mot sturmien standard admet une unique représentation de la forme :

$$w = \lim_{n\to\infty} L_a^{d_1} L_b^{d_2} L_a^{d_3} L_b^{d_4} \dots L_a^{d_{2n-1}} L_b^{d_{2n}}(a)$$

où $d_1 \geq 0$ et $d_k \geq 1$ pour tout $k \geq 2$.

Pour terminer cette section, rappelons le théorème suivant présentant les relations entre les morphismes sturmiens :

1. Remarque : cette représentation n'est pas exprimée sous cette forme dans [7], dans laquelle $w = \lim_{n\to\infty} T^{c_1} L_a^{d_1} T^{c_2} L_b^{d_2} \dots T^{c_{2n-1}} L_a^{d_{2n-1}} T^{c_{2n}} L_b^{d_{2n}}(a)$ où T est le décalage défini, pour tout mot infini $(a_n)_{n\geq 0}$ où a_n est une lettre pour tout $n \geq 0$, par $T(a_n)_{n\geq 0} = (a_{n+1})_{n\geq 0}$. On peut vérifier que pour les entiers c, d tels que $d \geq c \geq 0$ et pour tout mot infini w, $T^c L_a^d(w) = L_a^{d-c} R_a^c(w)$ et $T^c L_b^d(w) = L_b^{d-c} R_b^c(w)$.

Théorème 2.2.2. [59] *Le monoïde $\{L_a, L_b, R_a, R_b, E\}^*$ des morphismes sturmiens admet la présentation suivante :*

(1) $EE = Id$,

(2) $EL_a = L_bE$ *et* $ER_a = R_bE$,

(3) $L_aL_b^nR_a = R_aR_b^nL_a$, *pour tout* $n \geq 0$.

Par les relations (2) et (3), on obtient $L_bL_a^nR_b = R_bR_a^nL_b$ pour tout $n \geq 0$.

2.3 Mots et morphismes épisturmiens

Parmi les généralisations des mots sturmiens proposées sur un alphabet quelconque, deux sont très proches : les suites d'Arnoux-Rauzy (voir par exemple [4, 50, 82, 91]) et les *mots épisturmiens* (voir par exemple [28, 49, 51, 34]), la première catégorie étant une sous-classe de la seconde. Plus précisément, la famille des mots épisturmiens est composée des suites d'Arnoux-Rauzy, de leur image par des morphismes épisturmiens, et de certains mots infinis périodiques. Dans le cas binaire, les suites d'Arnoux-Rauzy correspondent exactement aux mots sturmiens, alors que la famille des mots épisturmiens inclut tous les mots équilibrés récurrents (c'est-à-dire les mots équilibrés périodiques ainsi que les mots sturmiens) [35, 78, 89].

Soient deux lettres x et y, le morphisme d'*échange*, ou de *permutation*, de x et y est le morphisme défini sur $\mathcal{A}$ par :

$$E_{xy} : \begin{cases} x & \mapsto & y, \\ y & \mapsto & x, \\ z & \mapsto & z, \forall z \notin \{x, y\} \end{cases}$$

On note $Exch(\mathcal{A})$ l'ensemble des morphismes de permutation incluant l'identité, c'est-à-dire les morphismes f tels que $f(\mathcal{A}) = \mathcal{A}$.

On appelle *morphismes épisturmiens* les composés des morphismes de $Exch(A)$ et des morphismes L_a et R_a où, pour tout $a \in \mathcal{A}$:

$$L_a : \begin{cases} a & \mapsto & a \\ b & \mapsto & ab \end{cases}, \quad R_a : \begin{cases} a & \mapsto & a \\ b & \mapsto & ba \end{cases} \quad \text{pour tout } b \neq a \text{ dans } \mathcal{A}.$$

Ces morphismes sont précisément ceux qui préservent l'ensemble des mots épisturmiens apériodiques [28]. Appliquer le morphisme L_a (resp. R_a) sur un mot revient à placer une occurrence de la lettre a sur la gauche (resp. sur la droite) de chaque occurrence de toute lettre différente de a.

Les morphismes épisturmiens sont dits *purs* lorsqu'ils sont obtenus par composition des éléments des ensembles :

$$\mathcal{L}_\mathcal{A} = \{L_a \mid a \in \mathcal{A}\} \quad \text{et} \quad \mathcal{R}_\mathcal{A} = \{R_a \mid a \in \mathcal{A}\}.$$

Les *morphismes épistandard* sont les morphismes obtenus par composition des morphismes de $\mathcal{L}_\mathcal{A}$ et des permutations de $\mathcal{A}$. De même que pour les morphismes épisturmiens purs, les *morphismes épistandard purs* sont ceux obtenus par composition des morphismes dans $\mathcal{L}_\mathcal{A}$.

Notons que les morphismes épisturmiens sont exactement les morphismes sturmiens quand $\mathcal{A}$ est un alphabet binaire.

Dans [88], Richomme a donné une présentation du monoïde des morphismes épisturmiens (indépendamment traité dans [51]) :

Théorème 2.3.1. *[88]*

Le monoide des morphismes épisturmiens avec $Exch(\mathcal{A}) \cup \{L_\alpha, R_\alpha \mid \alpha \in \mathcal{A}\}$ *comme ensemble de générateurs a la présentation suivante (x, y, z, et t sont des lettres distinctes deux à deux) :*

$$\begin{array}{ll} E_{xy}E_{xy} & = Id, \\ E_{xy}E_{yz} & = E_{yz}E_{zx}, \\ E_{xy}E_{zt} & = E_{zt}E_{xy}, \\ E_{xy}L_\alpha & = L_\alpha E_{xy}, \forall \alpha \notin \{x, y\} \\ E_{xy}R & = RE_{xy}, \\ L_\alpha E_1 L_\alpha E_2 ... L_\alpha E_k R_\alpha & = R_\alpha E_1 R_\alpha E_2 ... R_\alpha E_k L_\alpha \end{array}$$

où $k \geq 1$ *est un entier et* $E_1, \ldots, E_k \in Exch(\mathcal{A})$ *sont tels que* $E_1 \ldots E_k(\alpha) = \alpha$ *et pour tout entier* i, $2 \leq i \leq k$, $E_i \ldots E_k(\alpha) \neq \alpha$.

On obtient la présentation suivante pour les morphismes épisturmiens purs :

Théorème 2.3.2. *(conséquence directe de [88]) Le monoïde des morphismes épisturmiens purs avec* $\{L_\alpha, R_\alpha \mid \alpha \in \mathcal{A}\}$ *comme ensemble de générateurs a la présentation suivante :*

$$R_{a_1} R_{a_2} \ldots R_{a_k} L_{a_1} = L_{a_1} L_{a_2} \ldots L_{a_k} R_{a_1}$$

où $k \geq 1$ *est un entier et* $a_1, \ldots, a_k \in \mathcal{A}$ *avec* $a_1 \neq a_i$ *pour tout* i, $2 \leq i \leq k$.

Justin et Pirillo [49] ont montré que tout mot épisturmien peut être *décomposé infiniment* sur l'ensemble des morphismes épisturmiens purs (en général, cette décomposition n'est pas unique, voir la partie 3). Quelques définitions et notations supplémentaires sont nécessaires pour présenter ce résultat.

Soit un nouvel alphabet $\bar{\mathcal{A}} = \{\bar{x} \mid x \in \mathcal{A}\}$. Une lettre $\bar{x}$ est considérée comme la lettre x avec un *spin* R, alors que x elle-même a un spin L.

Un mot fini ou infini sur $\mathcal{A} \cup \bar{\mathcal{A}}$ est appelé un mot *spinné*[2]. Dans ce qui suit, un mot possédant uniquement des lettres de spin L (resp. spin R) sera appelé un mot de spin L ou L-spinné (resp. de spin R ou R-spinné). L'*opposé* $\bar{w}$ d'un mot fini ou infini spinné w est obtenu en échangeant tous les spins dans w. Par exemple, si $v = ab\bar{a}$, alors $\bar{v} = \bar{a}\bar{b}a$. Quand $v \in \mathcal{A}^+$, alors son opposé $\bar{v} \in \bar{\mathcal{A}}^+$ est un mot de spin R et on pose $\bar{\varepsilon} = \varepsilon$. Pour un mot fini ou infini $w = w_1 w_2 \ldots$ sur $\mathcal{A}$, on note parfois $\breve{w} = \breve{w}_1 \breve{w}_2 \cdots$ tout mot spinné tel que $\breve{w}_i = w_i$ si $\breve{w}_i$ est de spin L et $\breve{w}_i = \bar{w}_i$ si $\breve{w}_i$ est de spin R. Un tel mot $\breve{w}$ est appelé une *version spinnée* de w. On notera également parfois $\hat{w}$ une version spinnée d'un mot w différente de $\breve{w}$.

Les mots spinnés sont étroitement liés aux morphismes épisturmiens. Pour $a \in \mathcal{A}$, soit $\mu_a = L_a$ et $\mu_{\bar{a}} = R_a$. L'opérateur μ peut être naturellement étendu à un morphisme du monoïde libre $(\mathcal{A} \cup \bar{\mathcal{A}})^*$ vers un morphisme épisturmien : pour un mot spinné fini $\breve{w} = \breve{w}_1 \ldots \breve{w}_n$ sur $\mathcal{A} \cup \bar{\mathcal{A}}$, $\mu_{\breve{w}} = \mu_{\breve{w}_1} \ldots \mu_{\breve{w}_n}$ (μ_ε est le morphisme identité). On dit que le mot w *dirige* ou est un *mot directeur* du morphisme μ_w.

Exemple 2.3.3. Le mot $w = w_1 w_2 w_3 w_4 w_5 = \bar{a}\bar{b}ab\bar{a}$ dirige le morphisme $R_a R_b L_a L_b R_a$. Les lettres w_1, w_2, et w_5 sont de spin R, et les lettres w_3 et w_4 sont de spin L.

2. N'ayant pas trouvé de traduction satisfaisante au mot anglais *spinned* de la littérature, j'ai choisi ce terme pour désigner les mots dont les lettres ont un spin.

Justin et Pirillo étendent la notion de mots directeurs aux mots épisturmiens infinis :

Théorème 2.3.4. [49]

i) Un mot infini $\mathbf{s} \in \mathcal{A}^\omega$ *est épistandard si et seulement s'il existe un mot infini* $\Delta = x_1x_2x_3\cdots$ *sur* $\mathcal{A}$ *et une suite infinie* $(\mathbf{s}^{(n)})_{n\geq 0}$ *de mots infinis tels que* $\mathbf{s}^{(0)} = \mathbf{s}$ *et pour tout* $n \geq 1$, $\mathbf{s}^{(n-1)} = \mu_{x_n}(\mathbf{s}^{(n)})$ *(i.e.* $\mathbf{s}^{(n-1)} = L_{x_n}(\mathbf{s}^{(n)})$*).*

ii) Un mot infini $\mathbf{t} \in \mathcal{A}^\omega$ *est épisturmien si et seulement s'il existe un mot infini spinné* $\breve{\Delta} = \breve{x}_1\breve{x}_2\breve{x}_3\cdots$ *sur* $\mathcal{A}\cup\bar{\mathcal{A}}$ *et une suite infinie* $(\mathbf{t}^{(n)})_{n\geq 0}$ *de mots infinis récurrents tels que* $\mathbf{t}^{(0)} = \mathbf{t}$ *et pour tout* $n \geq 1$, $\mathbf{t}^{(n-1)} = \mu_{\breve{x}_n}(\mathbf{t}^{(n)})$*.*

Pour tout mot épistandard (resp. épisturmien) $\mathbf{t}$ et tout mot infini Δ de spin L (resp. tout mot spinné $\breve{\Delta}$) satisfaisant les conditions du théorème ci-dessus, on dit que Δ (resp. $\breve{\Delta}$) est un *mot directeur (spinné)* de $\mathbf{t}$ ou que $\mathbf{t}$ est *dirigé par* Δ (resp. $\breve{\Delta}$). Ainsi, dans le cas *ii)* du théorème 2.3.4, tout $\mathbf{t}^{(n)}$ est un mot épisturmien dirigé par $\breve{x}_{n+1}\breve{x}_{n+2}\cdots$.

La présentation du monoïde des morphismes épisturmiens purs peut être exprimée en termes d'équivalence de blocs sur les mots directeurs. Un mot de la forme xvx, où $x \in \mathcal{A}$ et $v \in (\mathcal{A}\setminus\{x\})^*$, est appelé un *bloc* basé sur x. Une *transformation* d'un bloc basé sur x est le remplacement dans un mot spinné d'une occurrence de $xv\bar{x}$ (où xvx est un bloc) par $\bar{x}\bar{v}x$ ou vice-versa. Deux mots finis spinnés w, w' sont dits *équivalents par blocs* si l'on peut passer de l'un à l'autre par une chaîne (éventuellement vide) de transformations de bloc, et dans ce cas on écrit $w \equiv w'$. Par exemple, $\bar{b}\bar{a}b\bar{c}b\bar{a}\bar{c}$ et $babc\bar{b}\bar{a}\bar{c}$ sont équivalents par blocs car $\bar{b}\bar{a}b\bar{c}b\bar{a}\bar{c} \to ba\bar{b}\bar{c}b\bar{a}\bar{c} \to babc\bar{b}\bar{a}\bar{c}$ et vice-versa. L'équivalence par blocs est une relation d'équivalence sur les mots spinnés, et de plus si $w \equiv w'$ alors w et w' sont des versions spinnées du même mot sur $\mathcal{A}$.

Ainsi, le théorème 2.3.2 peut être reformulé en termes d'équivalence par blocs : soient w, w' deux mots spinnés sur $\mathcal{A}\cup\bar{\mathcal{A}}$, alors $\mu_w = \mu_{w'}$ si et seulement si $w \equiv w'$.

Le résultat suivant de Justin et Pirillo précise la forme des mots spinnés infinis dirigeant un unique mot épisturmien :

Proposition 2.3.5. [49, Prop. 3.11] [3]

1. *Tout mot infini spinné* $\breve{\Delta}$ *possédant une infinité de lettres de spin L dirige un unique mot épisturmien commençant par la lettre de spin L la plus à gauche dans* $\breve{\Delta}$*.*
2. *Tout mot infini* $\breve{\Delta}$ *de spin R dirige exactement* $|\mathrm{Ult}(\Delta)|$ *mots épisturmiens.*
3. *Soit* $\breve{\Delta}$ *un mot infini de spin R, et soit a une lettre telle que* $\bar{a} \in \mathrm{Ult}(\breve{\Delta})$*. Alors* $\breve{\Delta}$ *dirige exactement un mot épisturmien commençant par a.*

Cependant, en général, la décomposition des mots épisturmiens n'est pas unique, c'est-à-dire qu'il peut exister plusieurs mots directeurs équivalents pour un même mot épisturmien. Ceci nous a amenés à caractériser ces équivalences et à proposer une forme normalisée permettant de simplifier l'étude des propriétés des mots épisturmiens. C'est ce qui est présenté dans la partie suivante.

3. Dans [49], l'item 3 a été établi dans le cas plus général où $\breve{\Delta}$ est ultimement de spin R. Dans ce cas, $\breve{\Delta}$ dirige exactement un mot épisturmien pour chaque lettre $\bar{a}$ de $\mathrm{Ult}(\breve{\Delta})$, mais rien ne peut être dit sur la première lettre de ce mot.

Chapitre 3

Mots directeurs équivalents pour les mots épisturmiens

Les travaux décrits dans ce chapitre ont été réalisés et publiés en collaboration avec Gwénaël Richomme, actuellement Professeur à l'Université de Montpellier, LIRMM et Amy Glen, actuellement à l'Université de Murdoch à Perth, Australie [LR2007a, GLR2008a, GLR2009]. Dans ces travaux, nous avons caractérisé les paires de mots dirigeant le même mot épisturmien, nous avons proposé une manière unique de définir un mot épisturmien grâce à la normalisation de ses mots directeurs, et enfin nous avons caractérisé les mots épisturmiens possédant un unique mot directeur.

Comme nous l'avons vu dans la section précédente, Justin et Pirillo ont montré de nombreux résultats sur les mots directeurs équivalents pour les mots épisturmiens, c'est-à-dire pour les mots infinis dirigeant le même mot épisturmien, cependant ils n'ont pas systématiquement fourni les formes de ces mots équivalents. Nous avons complété leurs résultats par la caractérisation suivante :

Théorème 3.0.1. *Soient deux mots infinis spinnés Δ_1 et Δ_2, les assertions suivantes sont équivalentes.*

i) Δ_1 *et* Δ_2 *dirigent un même mot épisturmien infini à droite ;*

ii) Δ_1 *et* Δ_2 *dirigent un même mot épisturmien bi-infini ;*

iii) L'un des cas suivants est vérifié pour i, j *tels que* $\{i, j\} = \{1, 2\}$ *:*

1. $\Delta_i = \prod_{n\geq 1} v_n$, $\Delta_j = \prod_{n\geq 1} z_n$ *où* $(v_n)_{n\geq 1}, (z_n)_{n\geq 1}$ *sont des mots spinnés tels que* $\mu_{v_n} = \mu_{z_n}$ *pour tout* $n \geq 1$ *et* $\prod$ *est le produit de concaténation ;*
2. $\Delta_i = wx \prod_{n\geq 1} v_n \breve{x}_n$, $\Delta_j = w'\bar{x} \prod_{n\geq 1} \bar{v}_n \hat{x}_n$ *où* w, w' *sont des mots spinnés tels que* $\mu_w = \mu_{w'}$, x *est une lettre de spin* L, $(v_n)_{n\geq 1}$ *est une suite non vide de mots de spin* L *sans* x, *et* $(\breve{x}_n)_{n\geq 1}$, $(\hat{x}_n)_{n\geq 1}$ *sont des suites non vides de mots spinnés sur* $\{x, \bar{x}\}$ *tels que, pour tout* $n \geq 1$, $|\breve{x}_n| = |\hat{x}_n|$ *et* $|\breve{x}_n|_x = |\hat{x}_n|_x$ *;*
3. $\Delta_1 = w\mathbf{x}$ *et* $\Delta_2 = w'\mathbf{y}$ *où* w, w' *sont des mots spinnés,* x *et* y *des lettres, et* $\mathbf{x} \in \{x, \bar{x}\}^\omega$, $\mathbf{y} \in \{y, \bar{y}\}^\omega$ *sont des mots infinis spinnés tels que* $\mu_w(x) = \mu_{w'}(y)$.

Ainsi, tout mot épisturmien est dirigé par un mot infini spinné possédant une infinité de lettres de spin L, ainsi que par un mot spinné possédant à la fois une infinité de lettres de spin L et de lettres de spin R. De plus, si un mot épisturmien a deux mots directeurs satisfaisant les cas 2 et 3,

alors il a une infinité de mots directeurs. Remarquons que le dernier cas de ce théorème concerne uniquement des mots épisturmiens périodiques.

Exemple 3.0.2. Soient a, b, c trois lettres différentes de $\mathcal{A}$, les mots infinis spinnés $\Delta_1 = a(bc\bar{a})^\omega$ et $\Delta_2 = \bar{a}(\bar{b}\bar{c}\bar{a})^\omega$ dirigent un même mot épisturmien commençant par la lettre a. En effet, ces deux mots directeurs remplissent les conditions du cas 2 du théorème 3.0.1 avec $w = w' = \varepsilon$, $x = a$, et pour tout n, $v_n = bc$ et $\breve{x}_n = \hat{x}_n = \bar{a}$. De plus, le fait que Δ_1 commence par une lettre a de spin L implique que le mot qu'il dirige commence par a. De même $\Delta_1' = \bar{a}b(ca\bar{b})^\omega$ et $\Delta_2' = \bar{a}\bar{b}(\bar{c}\bar{a}\bar{b})^\omega$ dirigent un même mot épisturmien commençant par la lettre b. Remarque : puisque $\Delta_2 = \Delta_2'$, cela montre que la relation "dirige un même mot épisturmien" n'est pas une relation d'équivalence.

Exemple 3.0.3. Le mot périodique $(ab)^\omega = L_a(b^\omega) = R_b(a^\omega)$ est dirigé par ab^ω et $\bar{b}a^\omega$ ($L_a(b) = ab = R_b(a)$). Il est également dirigé par $(a\bar{b})^\omega$ (ainsi les lettres x et y peuvent être égales dans le cas 3 du théorème 3.0.1).

Remarque 3.0.4. Si un mot épisturmien $\mathbf{t}$ a deux mots directeurs satisfaisant les cas 2 ou 3, alors $\mathbf{t}$ a une infinité de mots directeurs. En effet, si le cas 2 est satisfait et que $\bar{x}$ apparaît dans $\breve{x}_p$ ($p \geq 1$), alors d'après le théorème 2.3.2, $x\left(\prod_{k=1}^{p-1} v_n\breve{x}_n\right) v_p\breve{x}_p'\bar{x} \equiv \bar{x}\left(\prod_{k=1}^{p-1} \bar{v}_n\breve{x}_n\right)\bar{v}_p\breve{x}_p'x$ où $\breve{x}_p'$ est tel que $\breve{x}_p \equiv \bar{x}\breve{x}_p'$. Ainsi $\mathbf{t}$ est également dirigé par $w\bar{x}\left(\prod_{k=1}^{p-1}\bar{v}_n\breve{x}_n\right)\bar{v}_p\breve{x}_p'x\prod_{n\geq p+1} v_n\breve{x}_n$. De même si le cas 2 est satisfait et que x apparaît dans $\breve{x}_p$ ($p \geq 1$), alors $\mathbf{t}$ est également dirigé par $w'x\left(\prod_{k=1}^{p-1} v_n\hat{x}_n\right)v_p\hat{x}_p'\bar{x}\prod_{n\geq p+1}\bar{v}_n\hat{x}_n$ où $\hat{x}_p'$ est tel que $\hat{x}_p \equiv x\hat{x}_p'$. Si le cas 3 est satisfait, alors $\mathbf{t}$ est périodique et dirigé par $w\mathbf{x}$ où $\mathbf{x}$ est une version spinnée de x^ω.

Les théorèmes 2.3.2 et 3.0.1 permettent de généraliser le théorème 2.2.1 aux mots épisturmiens :

Théorème 3.0.5. *Tout mot épisturmien* $\mathbf{t}$ *possède un mot directeur contenant une infinité de lettres de spin L, mais aucun facteur dans* $\bigcup_{a\in\mathcal{A}} \bar{a}\bar{\mathcal{A}}^*a$. *Un tel mot directeur est unique si* $\mathbf{t}$ *est apériodique.*

Un mot directeur d'un mot épisturmien apériodique $\mathbf{t}$ vérifiant la propriété ci-dessus est appelé *mot directeur normalisé* de $\mathbf{t}$. On étend cette définition aux morphismes ; un mot fini spinné w est dit *mot directeur normalisé* du morphisme μ_w si w n'a aucun facteur dans $\bigcup_{a\in\mathcal{A}} \bar{a}\bar{\mathcal{A}}^*a$.

On peut remarquer que d'après le théorème 2.3.2, pour tout morphisme dans $L_a\mathcal{L}_\mathcal{A}^*R_a$, on peut trouver une autre décomposition du morphisme dans l'ensemble $R_a\mathcal{R}_\mathcal{A}^*L_a$. De façon équivalente, pour tout mot spinné dans $a\mathcal{A}^*\bar{a}$, il existe un mot w' dans $\bar{a}\bar{\mathcal{A}}^*a$ tel que $\mu_w = \mu_{w'}$. À partir de cette idée on peut prouver le lemme suivant :

Lemme 3.0.6. *Tout morphisme épisturmien pur possède un unique mot directeur normalisé.*

La preuve du théorème 3.0.5 est basée sur une extension de ce lemme aux mots infinis.

Exemple 3.0.7. Soit f le morphisme épisturmien pur dirigé par $\bar{a}\bar{b}c\bar{b}a\bar{b}\bar{a}\bar{c}\bar{b}\bar{a}\bar{c}a$. D'après le théorème 2.3.2, $\mu_{\bar{a}\bar{c}\bar{b}\bar{a}\bar{c}a} = \mu_{\bar{a}\bar{c}\bar{b}ac\bar{a}} = \mu_{ac\bar{b}ac\bar{a}}$ et donc $\mu_{\bar{a}\bar{b}c\bar{b}a\bar{b}\bar{a}\bar{c}\bar{b}\bar{a}\bar{c}a} = \mu_{\bar{a}\bar{b}c\bar{b}a\bar{b}ac\bar{b}ac\bar{a}}$ et $\bar{a}\bar{b}c\bar{b}a\bar{b}ac\bar{b}ac\bar{a}$ est le mot directeur normalisé de f.

De plus, nous avons caractérisé les mots épisturmiens ayant un unique mot directeur :

Théorème 3.0.8. *Un mot épisturmien possède un unique mot directeur si et seulement si son mot directeur normalisé contient :*

1) *une infinité de lettres de spin L,*
2) *une infinité de lettres de spin R,*
3) *aucun facteur dans* $\bigcup_{a\in\mathcal{A}} \bar{a}\bar{\mathcal{A}}^* a$,
4) *et aucun facteur dans* $\bigcup_{a\in\mathcal{A}} a\mathcal{A}^*\bar{a}$.

Un tel mot épisturmien est nécessairement apériodique.

Par exemple, une famille particulière de mots épisturmiens possédant un unique mot directeur est celle dont les mots directeurs sont des mots infinis possédant à la fois une infinité de lettres de spin L et de lettres de spin R, qui de plus soient *réguliers*, c'est-à-dire tels qu'une lettre apparaît toujours avec le même spin dans le mot directeur (dans la littérature, on parle de *regular wavy words*).

Dans le cas sturmien, on a :

Proposition 3.0.9. *Tout mot sturmien possède un unique mot directeur spinné ou une infinité de mots directeurs spinnés. De plus, un mot sturmien possède un unique mot directeur si et seulement si son mot directeur normalisé est un mot régulier possédant à la fois une infinité de lettres de spin L et de lettres de spin R.*

Chapitre 4

Quasipériodicité des mots et morphismes sturmiens

Les résultats de cette section ont été publiés en collaboration avec Gwénaël Richomme dans [LR2007b]. En particulier, le théorème 4.2.6 caractérise tous les mots sturmiens non-quasipériodiques.

Le problème qui nous intéresse ici est celui de la structure quasipériodique des mots. Commençons par rappeler quelques notions concernant la quasipériodicité, sur les mots finis et infinis, puis sur les morphismes.

4.1 Mots et morphismes quasipériodiques

Nous utilisons les définitions de [3]. Un mot u *couvre* un autre mot w si pour tout $i \in \{1, \ldots, |w|\}$, il existe $j \in \{1, \ldots, |u|\}$ tel qu'il y ait une occurrence de u commençant à la position $i - j + 1$ dans le mot w. Lorsque $u \neq w$, on dit que u est une *quasipériode* de w et que w est *quasipériodique*. De façon naturelle, un mot qui n'est pas quasipériode est simplement appelé mot *non quasipériodique*, ou encore mot *superprimitif* (Marcus [62] appelle de tels mots des mots *minimaux*). On observe qu'aucun mot de longueur 1 n'est quasipériodique. Le mot

$$w = abaababaabaababaaba$$

possède les quasipériodes aba, $abaaba$ et $abaababaaba$. Seule aba est superprimitive. Plus généralement dans [3], il est prouvé que tout mot fini quasipériodique possède exactement une quasipériode superprimitive. C'est une conséquence du fait que toute quasipériode d'un mot fini w est un bord propre de w.

Lorsque l'on définit les mots infinis quasipériodiques, au lieu de considérer les indices de début d'occurrences d'une quasipériode, nous avons choisi pour simplifier la suite de considérer les mots précédant les occurrences d'une quasipériode. Un mot infini $\mathbf{w}$ est *quasipériodique* s'il existe un mot fini u et des mots $(p_n)_{n\geq 0}$ tels que $p_0 = \varepsilon$ et, pour $n \geq 0$, $0 < |p_{n+1}| - |p_n| \leq |u|$ et $p_n u$ est un préfixe de $\mathbf{w}$. On dit que u *couvre* $\mathbf{w}$, ou que $\mathbf{w}$ est *u-quasipériodique*. Le mot u est aussi appelé une *quasipériode* et l'on dit que la suite $(p_n u)_{n\geq 0}$ est *une suite couvrante de préfixes du mot* $\mathbf{w}$.

Par exemple, le mot de Fibonacci, point fixe du morphisme $\varphi : a \mapsto ab$, $b \mapsto a$ est aba-quasipériodique. En fait $\varphi^\omega(a)$ a une infinité de quasipériodes superprimitives (voir [LR2004] pour

une caractérisation de toutes les quasipériodes de $\varphi^\omega(a)$), et [61] pour d'autres exemples de mots infinis quasipériodiques). Notons que pour tout entier positif n, il existe un mot infini ayant exactement n quasipériodes (comme par exemple le mot $(ab)^n a(ab)^\omega$)), ou ayant exactement n quasipériodes superprimitives [LR2004].

On observe que toute quasipériode d'un mot (fini ou infini) w est un préfixe de w. Ainsi, w possède une unique quasipériode de plus petite longueur, que nous appellerons *plus petite quasipériode* de w. Lorsque w est fini, la plus petite quasipériode de w est nécessairement sa quasipériode superprimitive. Lorsque w est infini, sa plus petite quasipériode est également superprimitive, mais il peut exister d'autres quasipériodes superprimitives (voir ci-dessus). De plus, on montre aisément le lemme suivant :

Lemme 4.1.1. *Si w est un mot infini quasipériodique de plus petite quasipériode u, alors uu est un facteur de w.*

Ce lemme 4.1.1 n'est pas vérifié pour les mots finis, comme le montre par exemple le mot aba-quasipériodique $ababa$.

Remarquons également le fait immédiat suivant :

Fait 4.1.2. Si w est un mot (fini ou infini) u-quasipériodique et f est un morphisme non-effaçant, alors $f(w)$ est $f(u)$-quasipériodique.

Notons que la réciproque de ce fait n'est pas vraie. Par exemple, soit $f = R_a$ et $\mathbf{w} = ab^\omega$, alors $R_a(\mathbf{w}) = a(ba)^\omega$. Le mot $R_a(\mathbf{w})$ est couvert par $R_a(ab)$, mais $\mathbf{w}$ n'est pas couvert par ab.

Nous pouvons maintenant aborder la partie suivante, qui présente le problème de la quasipériodicité sur les mots sturmiens.

4.2 Mots sturmiens non-quasipériodiques

Soit w un mot sturmien. Si l'on note n le plus petit nombre de a entre deux b dans w et par i le nombre initial de a dans w, on peut déduire de la propriété d'équilibre de w que w appartient à $a^i\{ba^n, ba^{n+1}\}^\omega$ avec $0 \leq i \leq n+1$. Quand $0 < i \leq n$, w appartient à $\{a^iba^{n-i}, a^iba^{n+1-i}\}^\omega$ et w est a^iba^{n-i+1}-quasipériodique (et a^iba^{n-i+1} est la plus petite quasipériode de w). Ainsi :

Fait 4.2.1. Si w est un mot sturmien non-quasipériodique, alors il existe un entier n tel que w appartient à $a^{n+1}b\{a^nb, a^{n+1}b\}^\omega \cup \{ba^n, ba^{n+1}\}^\omega$.

Bien sûr certains mots sturmiens dans $a^{n+1}b\{a^nb, a^{n+1}b\}^\omega \cup \{ba^n, ba^{n+1}\}^\omega$ sont quasipériodiques : c'est le cas des images de tout mot sturmien quasipériodique commençant par a par le morphisme sturmien $L_a^nR_b : a \mapsto a^{n+1}b, b \mapsto a^nb$.

Une conséquence du fait 4.2.1 est la suivante :

Lemme 4.2.2. *Pour tout mot sturmien w et $x \in \{a, b\}$, $L_xR_x(w) = R_xL_x(w)$ est quasipériodique.*

Démonstration. Sans perte de généralité, supposons $x = a$. D'après le théorème 2.2.2, $L_aR_a = R_aL_a$. Or $L_aR_a(a) = a$ and $L_aR_a(b) = aba$. D'après le fait 4.1.2, si w est quasipériodique, alors $L_aR_a(w)$ l'est aussi. Supposons maintenant que w est un mot sturmien non-quasipériodique. D'après le fait 4.2.1, w appartient à $a^{n+1}b\{a^nb, a^{n+1}b\}^\omega \cup \{ba^n, ba^{n+1}\}^\omega$ pour un entier n. Ainsi $L_aR_a(w)$ appartient à l'un des ensembles $a^{n+1}aba\{a^naba, a^{n+1}aba\}^\omega$ ou $\{abaa^n, abaa^{n+1}\}^\omega$. Donc $L_aR_a(w)$ est $a^{n+2}ba$-quasipériodique ou aba^{n+2}-quasipériodique. □

On remarque que ba^ω et $L_aR_a(ba^\omega) = aba^\omega$ ne sont pas quasipériodiques. Le lemme 4.2.2 n'est donc pas vérifié pour des mots quelconques (même s'ils sont équilibrés), au contraire du fait suivant qui est une conséquence directe de la définition de $L_aL_b : a \mapsto aba$, $b \mapsto ab$, et $L_bL_a : a \mapsto ba$, $b \mapsto bab$.

Fait 4.2.3. Pour tout mot infini w, $L_aL_b(w)$ est aba-quasipériodique et $L_bL_a(w)$ est bab-quasipériodique.

Le lemme 4.2.2 et le fait 4.2.3 sont utiles pour montrer que la condition du théorème 4.2.6 suivant est nécessaire. Pour montrer qu'elle est suffisante, il faut considérer les cas où l'image d'un mot par un morphisme sturmien n'est pas nécessairement quasipériodique, ce qui est fait par les deux lemmes suivants.

Lemme 4.2.4. *Soit $x \in \{a, b\}$ et soit w un mot équilibré commençant par x. Le mot $L_x(w)$ est quasipériodique si et seulement si w est quasipériodique. De plus, dans ce cas, la plus petite quasipériode de $L_x(w)$ est le mot $L_x(v)$ où v est la plus petite quasipériode de w.*

Lemme 4.2.5. *Soient x, y des lettres telles que $\{x, y\} = \{a, b\}$ et soit w un mot infini commençant par x. Le mot $R_y(w)$ est quasipériodique si et seulement si w est quasipériodique. De plus lorsque ces mots sont quasipériodiques, la plus petite quasipériode de $R_y(w)$ est le mot $R_y(v)$ où v est la plus petite quasipériode de w.*

On peut observer une différence entre les deux lemmes précédents : le lemme 4.2.4 concerne seulement les mots équilibrés, alors que le lemme 4.2.5 fonctionne pour des mots quelconques (commençant par x). Remarquons que le lemme 4.2.4 devient faux si l'on ne considère pas des mots équilibrés. En effet le mot $w = abab(aaab)^\omega$ n'est pas quasipériodique, alors que $L_a(w) = aabaabaa(aabaa)^\omega$ est $aabaa$-quasipériodique. Les deux lemmes deviennent également faux si l'on considère des mots sturmiens commençant par y où $\{x, y\} = \{a, b\}$. En effet, considérons le cas $x = a$, $y = b$: le mot $w = (L_bR_a)^\omega(a)$ n'est pas quasipériodique [LR2004] ; ce mot sturmien commence par b et le mot $L_a(w)$ (resp. $R_b(w)$) est aba-quasipériodique (resp. bab-quasipériodique).

On peut maintenant établir la caractérisation des mots sturmiens non-quasipériodique :

Théorème 4.2.6. *Un mot sturmien w n'est pas quasipériodique si et seulement s'il peut être infiniment décomposé sur $\{L_a, R_b\}$ ou sur $\{L_b, R_a\}$. En d'autres termes, un mot sturmien w n'est pas quasipériodique si et seulement si*

$$w = \lim_{n\to\infty} L_a^{d_1} R_b^{d_2} L_a^{d_3} R_b^{d_4} \dots L_a^{d_{2n-1}} R_b^{d_{2n}}(a)$$

ou

$$w = \lim_{n\to\infty} L_b^{d_1} R_a^{d_2} L_b^{d_3} R_a^{d_4} \dots L_b^{d_{2n-1}} R_a^{d_{2n}}(a)$$

où $d_k \geq 1$ pour tout $k \geq 2$ et $d_1 \geq 0$.

Démonstration. Prouvons d'abord que la condition est nécessaire. Soit w un mot sturmien non-quasipériodique. D'après le théorème 2.2.1,

$$w = \lim_{n\to\infty} L_a^{d_1-c_1} R_a^{c_1} L_b^{d_2-c_2} R_b^{c_2} \dots L_a^{d_{2n-1}-c_{2n-1}} R_a^{c_{2n-1}} L_b^{d_{2n}-c_{2n}} R_b^{c_{2n}}(a)$$

où $d_k \geq c_k \geq 0$ pour tout entier $k \geq 1$, $d_k \geq 1$ pour $k \geq 2$ et si $c_k = d_k$ alors $c_{k-1} = 0$.

D'après le lemme 4.2.2, pour $x \in \{a, b\}$ et pour tout mot sturmien w, $L_x R_x(w)$ est quasipériodique. D'après le fait 4.1.2, ceci implique que pour tout $k \geq 1$, $c_k = d_k$ ou $c_k = 0$. Supposons que $c_k = 0$ et $c_{k+1} = 0$ pour un entier $k \geq 1$. Alors $w = fL_aL_b(w')$ ou $w = fL_bL_a(w')$ pour un mot sturmien w' et un morphisme f. D'après le fait 4.2.3, w est quasipériodique. Donc pour tout $k \geq 1$, $c_k = 0$ implique $c_{k+1} = d_{k+1}$. On sait que pour tout $k \geq 2$, $c_k = d_k$ implique $c_{k-1} = 0$. Cela revient à dire que pour tout $k \geq 1$, $c_k \neq 0$ implique $c_{k+1} \neq d_{k+1}$. Mais alors pour tout k, $c_k = d_k$ ou $c_k = 0$. Ainsi $c_k = d_k$ implique $c_{k+1} = 0$, la condition est nécessaire.

Montrons maintenant que tout mot sturmien w qui peut être infiniment décomposé sur $\{L_a, R_b\}$ n'est pas quasipériodique (le cas $\{L_b, R_a\}$ est similaire). Supposons par contradiction que ce ne soit pas le cas. Soit $\mathcal{S}$ l'ensemble de tous les mots sturmiens w pouvant être décomposés sur $\{L_a, R_b\}$ et qui sont quasipériodiques. Soit u la plus petite quasipériode parmi toutes celles des mots de $\mathcal{S}$, et soit w un élément de $\mathcal{S}$ possédant u comme quasipériode. Par définition, $w = L_a(w')$ ou $w = R_b(w')$ pour un mot w' de $\mathcal{S}$. De plus d'après le théorème 2.2.1, $w = L_a^{d_1} R_b^{d_2} La(w_3)$ pour un mot sturmien w_3 et des entiers $d_1 \geq 0$, $d_2 \geq 1$: w commence par la lettre a. D'après les lemmes 4.2.4 et 4.2.5, $u = L_a(v)$ ou $u = R_b(v)$ où v est la plus petite quasipériode de w'. Puisque a^ω et b^ω ne sont pas des mots sturmiens (ils sont équilibrés mais pas non ultimement périodiques), $|v|_a \neq 0$ et $|v|_b \neq 0$. Par conséquent $|v| < |u|$. Ceci contredit le choix de u. D'où $\mathcal{S}$ est vide. □

Nous avons prouvé qu'il y a une connexion forte entre les mots de Lyndon et les mots sturmiens quasipériodiques :

Théorème 4.2.7. *Un mot sturmien w sur $\{a, b\}$ est non-quasipériodique si et seulement si w est un mot de Lyndon infini sur $\{a \prec b\}$ ou sur $\{b \prec a\}$.*

Notons que dans le cas des mots épisturmiens, la classe des mots épisturmiens non-quasipériodiques contient d'autres mots que des mots de Lyndon (voir [GLR2009] pour plus de détails).

Soit un mot w, notons $X(w)$ l'ensemble des mots infinis possédant le même ensemble de facteurs que w : $X(w)$ est invariant par opération de *décalage* (ou *shift*) et est appelé le sous-shift associé à w. Quand w est sturmien, il est connu (voir [7]) qu'un mot w' appartient à $X(w)$ si et seulement s'il est sturmien et que la suite $(d_k)_{k\geq 0}$ associée à sa décomposition d'après le théorème 2.2.1 est la même que celle impliquée dans la décomposition de w.

Pour terminer cette partie, observons que tout mot sturmien standard (c'est-à-dire un mot sturmien pouvant être décomposé en utilisant seulement L_a et L_b) est nécessairement quasipériodique. Cela fournit une nouvelle preuve à un résultat de T. Monteil [70, 69] : tout sous-shift sturmien contient un mot quasipériodique (mentionnons que le résultat de T. Monteil est plus précisément : tout sous-shift sturmien contient un mot quasipériodique multi-échelle, c'est-à-dire un mot possédant une infinité de quasipériodes). Le lemme 4.3.3 dans la section 4.3 permet de montrer que tout mot sturmien standard a une infinité de quasipériodes. Le théorème 4.2.6 montre également que dans tout sous-shift sturmien, il y a un mot non-quasipériodique.

4.3 Morphismes sturmiens quasipériodiques

On dit qu'un morphisme f est *sans quasipériode* si pour tout mot non-quasipériodique w, $f(w)$ est également non-quasipériodique. Un morphisme qui n'est pas sans quasipériode sera simplement

appelé *quasipériodique*. On remarque que tous les morphismes sturmiens (sauf E et Id) sont quasipériodiques. Pour le vérifier, il est suffisant de montrer que L_a, L_b, R_a et R_b sont quasipériodiques. Pour L_a et R_a (les cas L_b sont R_b similaires) on a : aba^ω et ab^ω sont non-quasipériodiques, cependant $L_a(aba^\omega) = aba(ab)^\omega$ et $R_a(ab^\omega) = a(ba)^\omega$ sont aba-quasipériodiques.

Dans la partie précédente, on a rencontré deux différents types de morphismes sturmiens (lemme 4.2.2 et fait 4.2.3). Le morphisme L_aL_b transforme tout mot en un mot quasipériodique, alors qu'il existe un mot non-quasipériodique w tel que $L_aR_a(w)$ n'est pas quasipériodique. En généralisant ces deux exemples, on observe que l'ensemble des morphismes quasipériodiques peut être partitionné de la façon suivante :

1. Un morphisme f sur A est appelé *fortement quasipériodique* (resp. sur un sous-ensemble X de A^ω) si pour tout mot infini non-quasipériodique w (resp. $w \in X$), $f(w)$ est quasipériodique.
2. Un morphisme f sur A est appelé *faiblement quasipériodique* (resp. sur un sous-ensemble X de A^ω) s'il existe deux mots infinis non-quasipériodiques w, w' (resp. $w, w' \in X$) tels que $f(w)$ est quasipériodique et $f(w')$ n'est pas quasipériodique.

Le but de cette partie est de répondre aux deux questions suivantes :

— Quels sont les morphismes sturmiens fortement (resp. faiblement) quasipériodiques ?

— Quels sont les morphismes sturmiens fortement (resp. faiblement) quasipériodiques sur l'ensemble des mots sturmiens ?

On remarque que ces deux questions ont des réponses différentes. En effet, comme le montre le lemme 4.2.2, L_aR_a est fortement quasipériodique sur les mots sturmiens, cependant nous avons vu que $L_aR_a(ba^\omega)$ n'est pas quasipériodique. Bien sûr, tout morphisme sturmien fortement quasipériodique est fortement quasipériodique sur les mots sturmiens, ou de façon équivalente, puisqu'un morphisme sturmien est quasipériodique, tout morphisme sturmien faiblement quasipériodique sur les mots sturmiens est faiblement quasipériodique.

Morphismes sturmiens faiblement et fortement quasipériodiques Dans cette section, on caractérise les morphismes sturmiens faiblement quasipériodiques (ce qui revient à caractériser les morphismes sturmiens fortement quasipériodiques puisque tout morphisme sturmien est faiblement ou fortement quasipériodique).

Proposition 4.3.1. *Un morphisme sturmien est faiblement quasipériodique si et seulement s'il appartient à l'ensemble*

$$\{E, Id\}\{L_a, R_b\}^*\{L_a, R_a\}^* \cup \{E, Id\}\{L_b, R_a\}^*\{L_b, R_b\}^*.$$

La preuve est une conséquence des deux lemmes suivants.

Lemme 4.3.2. *Soit f un morphisme de $\{L_a, L_b, R_a, R_b\}^*$ différent de l'identité. Le morphisme f appartient à $\{L_a, R_b\}^*\{L_a, R_a\}^* \cup \{L_b, R_a\}^*\{L_b, R_b\}^*$ si et seulement si f ne peut pas être écrit $f = f_1f_2f_3$ avec $f_1, f_3 \in \{L_a, L_b, R_a, R_b\}^*$ et tel que f_2 vérifie l'un des quatre propriétés suivantes :*

1. *$f_2 \in L_a\{L_a, L_b, R_a, R_b\}^*L_b \cup L_b\{L_a, L_b, R_a, R_b\}^*L_a$, ou*
2. *$f_2 = R_agL_a$ avec $g \notin \{R_a, L_a\}^*$ ou $f_2 = R_bgL_b$ avec $g \notin \{R_b, L_b\}^*$, or*
3. *$f_2 \in R_aR_b^+R_a$ ou $f_2 \in R_bR_a^+R_b$, ou*
4. *$f_2 \in R_a^+L_a^+R_b = L_a^+R_a^+R_b$ ou $f_2 \in R_b^+L_b^+R_a = L_b^+R_b^+R_a$.*

Lemme 4.3.3. *Tout morphisme f dans $L_a\{L_a, L_b, R_a, R_b\}^* L_b \cup L_b\{L_a, L_b, R_a, R_b\}^* L_a$ est fortement quasipériodique.*

Une approche pour montrer ce lemme est d'observer que $L_aR_b(w)$ ($L_aR_b(a) = aab$, $L_aR_b(b) = ab$) est aba-quasipériodique pour tout mot infini w commençant par b, et donc que tout morphisme de la forme $L_aR_bfL_b$ avec $f \in \{L_a, R_a, R_b\}^*$ est fortement quasipériodique.

Lemme 4.3.4. *Tout morphisme $f = R_agL_a$ avec $g \notin \{R_a, L_a\}^*$ ou $f = R_bgL_b$ avec $g \notin \{R_b, L_b\}^*$ est fortement quasipériodique.*

On peut en effet observer que $R_aR_b(w)$ ($R_aR_b(a) = aba$, $R_aR_b(b) = ba$) est aba-quasipériodique pour tout mot infini w commençant par a, et en déduire que tout morphisme de la forme $R_aR_bfL_a$ avec $f \in \{L_a, R_a, R_b\}^*$ est fortement quasipériodique.

De même on peut montrer que :

Lemme 4.3.5. *Tout morphisme f dans $R_aR_b^+R_a \cup R_bR_a^+R_b$ est fortement quasipériodique.*

et

Lemme 4.3.6. *Tout morphisme f dans $R_a^+L_a^+R_b = L_a^+R_a^+R_b$ ou dans $R_b^+L_b^+R_a = L_b^+R_b^+R_a$ est fortement quasipériodique.*

Preuve de la proposition 4.3.1.

D'après le théorème 2.2.2, $EL_a = L_bE$ et $ER_a = R_bE$, donc tout morphisme sturmien peut s'écrire fg avec $f \in \{Id, E\}$ et $g \in \{L_a, L_b, R_a, R_b\}^*$. Donc la proposition 4.3.1 est une conséquence de la suivante : un morphisme f dans $\{L_a, L_b, R_a, R_b\}^*$ est faiblement quasipériodique si et seulement si f appartient à l'ensemble $X = \{L_a, R_b\}^*\{L_a, R_a\}^* \cup \{L_b, R_a\}^*\{L_b, R_b\}^*$.

Pour le prouver, supposons d'abord que $f \in \{L_a, L_b, R_a, R_b\}^*$ est faiblement quasipériodique. On peut montrer que dans ce cas, ce morphisme ne peut pas s'écrire $f = f_1f_2f_3$ avec f_2 un morphisme fortement quasipériodique. D'où d'après les lemmes 4.3.2, 4.3.3, 4.3.4, 4.3.5 et 4.3.6, f appartient à X.

Supposons maintenant que $f \in X$. Puisque f est sturmien, il est quasipériodique et donc il suffit de montrer l'existence d'un mot w tel que $f(w)$ n'est pas quasipériodique. On le fait pour $f \in \{L_a, R_b\}^*\{L_a, R_a\}^*$ (l'autre cas est similaire). Il existe des morphismes $g \in \{L_a, R_b\}^*$ et $h \in \{L_a, R_a\}^*$ tels que $f = gh$. On peut vérifier que $h(aba^\omega) = a^nba^\omega$ pour un entier $n \geq 1$, et donc c'est un mot équilibré. D'après les lemmes 4.2.4 et 4.2.5, on déduit alors que $(g(h(aba^\omega)) = f(aba^\omega)$ n'est pas quasipériodique. □

Morphismes sturmiens faiblement quasipériodiques sur les mots sturmiens. La proposition 4.3.1 et le lemme 4.2.2 montrent que certains morphismes, comme par exemple L_aR_a, sont faiblement quasipériodiques alors qu'ils sont fortement quasipériodiques sur les mots sturmiens. Cette partie permet de caractériser ces morphismes. Rappelons que puisqu'un morphisme sturmien est quasipériodique, tout morphisme sturmien est faiblement ou fortement quasipériodique sur les mots sturmiens.

Proposition 4.3.7. *Un morphisme sturmien différent de E et Id est faiblement quasipériodique sur les mots sturmiens si et seulement s'il appartient à $\{E, Id\}\{L_a, R_b\}^* \cup \{E, Id\}\{L_b, R_a\}^*$.*

Démonstration. Supposons que $f \in \{E, Id\}\{L_a, R_b\}^* \cup \{E, Id\}\{L_b, R_a\}^*$. Sans perte de généralité, on peut supposer $f \in \{L_a, R_b\}^* \cup \{L_b, R_a\}^*$. Si f appartient à $\{L_a, R_b\}^*$ (resp. à $\{L_b, R_a\}^*$), en utilisant le théorème 4.2.6 on observe que $f((L_aR_b)^\omega)$ (resp. $f((L_bR_a)^\omega)$) n'est pas quasipériodique. Puisque tout morphisme sturmien est quasipériodique, f est faiblement quasipériodique sur les mots sturmiens.

Maintenant supposons que f est faiblement quasipériodique sur les mots sturmiens. D'après le théorème 2.2.2(2), $f \in \{E, Id\}\{L_a, L_b, R_a, R_b\}^*$. Remarquons qu'un morphisme f est faiblement quasipériodique sur les mots sturmiens si et seulement si Ef est faiblement quasipériodique sur les mots sturmiens. Sans perte de généralité, on peut donc supposer que f appartient à $\{L_a, L_b, R_a, R_b\}^*$ et montrer que $f \in \{L_a, R_b\}^* \cup \{L_b, R_a\}^*$. D'après la proposition 4.3.1, f appartient à $\{L_a, R_b\}^*\{L_a, R_a\}^* \cup \{L_b, R_a\}^*\{L_b, R_b\}^*$. Supposons par contradiction que $f \notin \{L_a, R_b\}^* \cup \{L_b, R_a\}^*$. L'un des quatre cas suivants est vérifié :

1. $f = gL_aR_a$ avec $g \in \{L_a, R_b\}^*\{L_a, R_a\}^*$;
2. $f = gR_bR_a^i$ avec $g \in \{L_a, R_b\}^*$, $i \geq 1$;
3. $f = gL_bR_b$ avec $g \in \{L_b, R_a\}^*\{L_b, R_b\}^*$;
4. $f = gR_aR_b^i$ avec $g \in \{L_b, R_a\}^*$, $i \geq 1$.

Dans tous les cas, les résultats précédents permettent de montrer que $f(w)$ est quasipériodique pour tout mot sturmien non-quasipériodique w, et donc pour tout mot sturmien (d'après le fait 4.1.2). Donc f est fortement quasipériodique sur les mots sturmiens. Ceci est une contradiction, donc $f \in \{L_a, R_b\}^* \cup \{L_b, R_a\}^*$. □

Chapitre 5

Quasipériodicité des mots et morphismes épisturmiens

Les résultats présentés dans ce chapitre ont été publiés en collaboration avec Amy Glen et Gwénaël Richomme dans [LR2007a, GLR2008b]. Les preuves de l'article étant plutôt techniques et nécessitant de nombreux ressorts et lemmes intermédiaires, j'ai choisi de présenter ici seulement les résultats principaux ainsi que des exemples permettant d'avoir une compréhension de la forme des mots épisturmiens quasipériodiques ou non.

5.1 Mots épisturmiens quasipériodiques

5.1.1 Clôture palindromique dans les mots épisturmiens

La *clôture palindromique (droite)* $w^{(+)}$ d'un mot fini w est le plus petit palindrome (unique) possédant w comme préfixe (voir [26]). C'est-à-dire, $w^{(+)} = wv^{-1}\widetilde{w}$ où v^{-1} est l'effacement du plus long suffixe palindromique v de w et $\widetilde{w}$ est le mot miroir de w. La fonction de *clôture palindromique itérée* [48], notée Pal, est définie récursivement par $Pal(\varepsilon) = \varepsilon$ et, pour tout mot w et toute lettre x, $Pal(wx) = (Pal(w)x)^{(+)}$. Par exemple, $Pal(abc) = (Pal(ab)c)^{(+)} = (abac)^{(+)} = abacaba$.

En généralisant une construction donnée dans [26] pour les *mots sturmiens standard*, Droubay, Justin and Pirillo ont établi la caractérisation suivante des mots épistandard :

Théorème 5.1.1. [28] *Un mot infini* $\mathbf{s} \in \mathcal{A}^\omega$ *est épistandard si et seulement s'il existe un mot infini* $\Delta = x_1x_2x_3\cdots$ *(*$x_i \in \mathcal{A}$*) tel que*

$$\mathbf{s} = \lim_{n\to\infty} Pal(x_1\cdots x_n).$$

Remarque : les palindromes $Pal(x_1\cdots x_n)$ sont très souvent notés u_{n+1} dans la littérature.

Dans [49], Justin and Pirillo ont montré que le mot Δ est exactement le mot directeur de $\mathbf{s}$ tel qu'il apparaît dans le théorème 2.3.4. De plus, par construction, Δ détermine *de façon unique* le mot épistandard $\mathbf{s}$. Par construction également, les mots $(Pal(x_1\cdots x_i))_{i\geq 0}$ sont exactement les préfixes palindromiques de $\mathbf{s}$.

Rappelons quelques relations existant entre les palindromes et les morphismes épisturmiens.

D'après [48, 49], on a $Pal(wv) = \mu_w(Pal(v))Pal(w)$ pour tous mots w, v. En particulier, pour toute lettre x, $Pal(xv) = L_x(Pal(v))x$ et $Pal(wx) = \mu_w(x)Pal(w)$.

Pour des lettres $(x_j)_{1\leq j\leq i}$, cette formule mène par récurrence à :

$$Pal(x_1\cdots x_i) = \mu_{x_1\cdots x_{i-1}}(x_i)\cdots\mu_{x_1}(x_2)x_1 = \prod_{1\leq j\leq i}\mu_{x_1\cdots x_{j-1}}(x_j)$$

où par convention, $x_1\cdots x_0 = \varepsilon$.

Si l'on considère $\breve{w} = \breve{x}_1\breve{x}_2\cdots\breve{x}_n$ une version spinnée de $w = x_1x_2\cdots x_n$ (vu comme préfixe d'une version spinnée $\breve{\Delta}$ de Δ), alors, pour tout mot fini v, on a :

$$\mu_{\breve{w}}(v) = S_{\breve{w}}^{-1}\mu_w(v)S_{\breve{w}} \quad \text{où } S_{\breve{w}} = \prod_{\substack{i=n,\ldots,1\\ |\breve{x}_i=\bar{x}_i}}\mu_{x_1\cdots x_{i-1}}(x_i).$$

Le mot $S_{\breve{w}}$ est appelé le *facteur de décalage* de $\mu_{\breve{w}}$ [51]. Observons que $S_{\breve{w}}$ est un préfixe de $Pal(w)$; en particulier, $S_{\bar{w}} = Pal(w)$ d'après l'équation précédente. Notons également que $\mu_{\breve{w}}(v) = \mathrm{T}^{|S_{\breve{w}}|}(\mu_w(v))$. Par exemple, pour $\breve{w} = a\bar{b}c\bar{a}$, on a $S_{\breve{w}} = \mu_{abc}(a)\mu_a(b) = abacabaab$. Ainsi puisque $\mu_{abca}(ca) = abacabaab.acabacaba$, $\mu_{a\bar{b}c\bar{a}}(ca) = \mathrm{T}^9(\mu_{abca}(ca)) = acabacaba.abacabaab$.

De même, pour tout mot infini $\mathbf{y}\in\mathcal{A}^\omega$, $\mu_{\breve{w}}(\mathbf{y}) = S_{\breve{w}}^{-1}\mu_w(\mathbf{y})$.

Cette formule utilisée avec $\breve{w} = \bar{w}$ montre que :

Fait 5.1.2. Tout mot de la forme $\mu_w(\mathbf{y})$ où $\mathbf{y}$ est infini commence par $Pal(w)$.

Notons également le résultat suivant, particulièrement utile dans l'étude de la quasipériodicité :

Proposition 5.1.3. *Supposons que* $\mathbf{s}$ *est un mot épistandard dirigé par* $\Delta = x_1x_2x_3\cdots$ *avec* $x_i\in\mathcal{A}$*, et soit* $\mathbf{t}$ *un mot épisturmien dirigé par une version spinnée de* Δ*. Alors* $\mathbf{t}$ *commence par* $Pal(x_1\cdots x_n)$ *pour un entier non négatif* n *si et seulement si* $\mathbf{t}$ *possède un mot directeur de la forme* $\breve{\Delta} = x_1\cdots x_n\breve{x}_{n+1}\breve{x}_{n+2}\cdots$ *où le préfixe* $x_1\cdots x_n$ *est de spin* L*.*

Démonstration. Quand $\mathbf{t}$ est dirigé par une version spinnée de Δ de la forme $\breve{\Delta} = x_1x_2\cdots x_n\breve{x}_{n+1}\breve{x}_{n+2}\cdots$ où le préfixe $x_1x_2\cdots x_n$ est de spin L, $\mathbf{t}$ commence par $Pal(x_1\cdots x_n)$ d'après le fait 5.1.2.

Réciproquement, supposons que $Pal(x_1\cdots x_n)$ est un préfixe de $\mathbf{t}$, et que $\mathbf{t}$ est dirigé par $\breve{\Delta} = (\breve{x}_i)_{i\geq 1}$ (une version spinnée de $\Delta = (x_i)_{i\geq 1}$). D'après le théorème 3.0.1, on peut supposer que $\breve{\Delta}$ contient une infinité de lettres de spin L. Si $n = 0$, il n'y a rien à prouver. Soit $x_1 = a$. Supposons d'abord que $\breve{x}_1 = \bar{a}$. Soit k le plus petit entier positif tel que $\breve{x}_k\in\mathcal{A}$. Puisque $\mathbf{t}$ commence par la lettre a (qui est la première lettre de $Pal(x_1\cdots x_n)$), on a $x_k = a$. Alors $\breve{\Delta} = \bar{a}\bar{x}_2\cdots\bar{x}_{k-1}a\breve{x}_{k+1}\cdots$, et donc d'après le théorème 2.3.2 $\mathbf{t}$ est également dirigé par le mot infini spinné commençant par $ax_2\cdots x_{j-1}\bar{a}$, avec $j\leq k$. Donc on peut supposer maintenant que $\breve{x}_1 = a$. Soit $\mathbf{t}'$ le mot épisturmien dirigé par $\mathrm{T}(\breve{\Delta}) = (\breve{x}_i)_{i\geq 2}$. On peut voir de par l'égalité $Pal(x_1\cdots x_n) = L_a(Pal(x_2\cdots x_n))a$ que $Pal(x_2\cdots x_n)$ est un préfixe de $\mathbf{t}'$. Donc par récurrence $\mathbf{t}'$ est dirigé par une version spinnée de $\mathrm{T}(\Delta)$ avec $x_2\cdots x_n$ comme préfixe. Et donc $\mathbf{t}$ est dirigé par une version spinnée de Δ de la forme $x_1x_2\cdots x_n\breve{x}_n\breve{x}_{n+1}\cdots$ où le préfixe $x_1x_2\cdots x_n$ est de spin L. □

5.1.2 Quasipériodicité des mots épisturmiens

Les "mots de retour" ont été introduits indépendamment dans [30] et [42] lors de l'étude des suites substitutives primitives. Ces mots sont définis de la façon suivante :

Définition 5.1.4. Soit v un facteur récurrent d'un mot infini $\mathbf{w} = w_1w_2w_3\cdots$, commençant aux positions $n_1 < n_2 < n_3\cdots$ dans $\mathbf{w}$. Alors tout mot $r_i = w_{n_i}w_{n_i+1}\cdots w_{n_{i+1}-1}$ est appelé *retour à* v dans $\mathbf{w}$.

C'est-à-dire, un retour à v dans $\mathbf{w}$ est un facteur non vide de $\mathbf{w}$ commençant à une occurrence de v et se terminant exactement avant l'occurrence suivante de v dans $\mathbf{w}$. Ainsi, si r est un retour à v dans $\mathbf{w}$, alors rv est un facteur de $\mathbf{w}$ contenant exactement deux occurrences de v, l'une comme préfixe et l'autre comme suffixe. Puisque tout mot épisturmien $\mathbf{t}$ est uniformément récurrent [28], tout facteur de $\mathbf{t}$ possède seulement un nombre fini de retours différents.

Note. Un retour à v dans $\mathbf{w}$ a toujours v comme préfixe ou est un préfixe de v. En particulier, on observe qu'un retour à v n'est pas nécessairement plus long que v, et alors v a des occurrences chevauchantes dans $\mathbf{w}$.

Ainsi, la définition suivante est équivalente à celle d'un mot infini quasipériodique en termes de mots de retour :

Lemme 5.1.5. *Un mot fini v est une quasipériode d'un mot infini $\mathbf{w}$ si et seulement si v est un préfixe récurrent de $\mathbf{w}$ tel que tout retour à v dans $\mathbf{w}$ est de longueur au plus $|v|$.*

En étudiant les mots de retour (voir [30] et [42] pour plus de détails) des facteurs des mots épistandard, on a montré que tout mot épistandard est quasipériodique. Plus précisément :

Théorème 5.1.6. *Supposons que $\mathbf{s}$ est un mot épistandard de mot directeur $\Delta = x_1x_2x_3\cdots$ avec $x_i \in \mathcal{A}$, et soit m le plus petit entier positif tel que $\mathrm{Alph}(x_1x_2\cdots x_m) = \mathrm{Alph}(\mathbf{s})$. Alors, pour tout $n \geq m$, $Pal(x_1\cdots x_n)$ est une quasipériode de $\mathbf{s}$.*

Exemple 5.1.7. Soit le mot de Tribonacci :

$$\mathbf{r} = abacabaabacababacabaabacabacabaabaca\cdots\ ,$$

qui est le mot épistandard dirigé par $(abc)^\omega$. Observons que $Pal(abc) = abacaba$ est le plus petit préfixe palindromique de $\mathbf{r}$ tel que $\mathrm{Alph}(Pal(abc)) = \mathrm{Alph}(\mathbf{r}) = \{a, b, c\}$. Les retours à $Pal(abc)$ dans $\mathbf{r}$ sont : $\mu_{abc}(a) = abacaba$, $\mu_{abc}(b) = abacab$, $\mu_{abc}(c) = abac$, aucun d'eux n'étant plus long que $Pal(abc)$. Donc $\mathbf{r}$ est $abacaba$-quasipériodique ; en fait $Pal(abc) = \mu_{abc}(a) = abacaba$ est la plus petite quasipériode de $\mathbf{r}$ puisque ses préfixes $abac$, $abaca$, $abacab$ ont des retours plus longs qu'eux-mêmes.

Autrement dit, les préfixes palindromiques suffisamment longs d'un mot épistandard $\mathbf{s}$ sont des quasipériodes de $\mathbf{s}$.

Pour étendre ce théorème aux mots épisturmiens, on procède de la façon suivante : appelons *quasipériode ultime* d'un mot infini $\mathbf{w}$ tout facteur v de $\mathbf{w}$ tel que v est quasipériode d'un suffixe de $\mathbf{w}$. On montre d'abord que tous les mots épisturmiens sont ultimement quasipériodiques. En caractérisant l'ensemble des quasipériodes ultimes $\mathcal{Q}$ d'un mot épistandard $\mathbf{s}$, puis en remarquant que les quasipériodes d'un mot épisturmien $\mathbf{t}$ dans le même sous-shift que $\mathbf{s}$ sont les mots de $\mathcal{Q}$ qui sont préfixes de $\mathbf{t}$, on en déduit les quasipériodes ultimes d'un mot épisturmien :

Lemme 5.1.8. *Supposons que* $\mathbf{s}$ *est un mot épistandard dirigé par* $\Delta \in \mathcal{A}^\omega$*. Alors l'ensemble des quasipériodes ultimes de tout mot épisturmien* $\mathbf{t}$ *dans le sous-shift de* $\mathbf{s}$ *est l'ensemble de tous les mots*

$$q \in F(Pal(wv)), \quad \text{avec } |q| \geq |\mu_w(Pal(v))|,$$

où w*,* v *sont des mots tels que* $\Delta = wv\mathbf{y}$ *avec* $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$ *et* $F(Pal(wv)$ *désigne l'ensemble des facteurs de* $Pal(wv)$*.*

En utilisant le résultat précédent sur les quasipériodes ultimes d'un mot épisturmien, nous avons pu décrire toutes les quasipériodes d'un mot épisturmien. C'est ce que propose le théorème suivant :

Théorème 5.1.9. *L'ensemble des quasipériodes d'un mot épisturmien* $\mathbf{t}$ *est l'ensemble de tous les mots*

$$\mu_{\breve{w}}(Pal(v))p, \quad \text{avec } p \text{ un préfixe de } S_{\breve{w}}^{-1}Pal(w), \tag{5.1}$$

où w*,* v *sont des mots de spin* L *tels que* $\mathbf{t}$ *est dirigé par* $\breve{w}v\breve{\mathbf{y}}$ *pour une version spinnée quelconque* $\breve{w}$ *de* w *et une version spinnée de* $\breve{\mathbf{y}}$ *d'un mot infini* $\mathbf{y}$ *de spin* L *avec* $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$*.*

De plus, la plus petite quasipériode de $\mathbf{t}$ *est le mot* $\mu_{\breve{w}}(Pal(v))$ *où* $|wv|$ *est la longueur minimale pour la propriété* $\mathrm{Alph}(wv) = \mathrm{Alph}(wv\mathbf{y})$*, et parmi toutes les décompositions de* wv *en* w *et* v*, le mot* v *est le plus petit suffixe de* wv *tel que* $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$*.*

Le théorème ci-dessus montre que s'il n'existe pas de mots $\breve{w}$, $\mathbf{y}$, $\breve{\mathbf{y}}$, et v comme définis ci-dessus tels que $\mathbf{t}$ est dirigé par $\breve{w}v\breve{\mathbf{y}}$ avec $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$, alors $\mathbf{t}$ n'a aucune quasipériode, et donc $\mathbf{t}$ n'est pas quasipériodique. Par exemple, tout mot $\breve{\Delta}$ régulier et comportant infiniment de lettres de spin L et de lettres de spin R (i.e. "regular wavy" comme défini dans la section 3) dirige clairement un mot épisturmien sans aucune quasipériode puisque $\breve{\Delta}$ est le seul mot directeur de $\mathbf{t}$ et qu'il ne contient pas de facteur v de spin L contenant toutes les lettres qui le suivent dans $\breve{\Delta}$. Par exemple, $(a\bar{b}c)^\omega$ dirige un mot épisturmien non-quasipériodique dans le sous-shift du mot de Tribonacci.

Illustrons maintenant la dernière partie du théorème. Pour un mot épistandard dirigé par $\Delta = ca(ab)^\omega$, i.e., l'image du mot de Fibonacci par le morphisme L_cL_a, le plus petit mot wv tel que $\Delta = wv\mathbf{y}$ pour un mot infini $\mathbf{y}$ avec $\mathrm{Alph}(wv) = \mathrm{Alph}(wv\mathbf{y})$ est le mot $caab$. Il y a trois façons de décomposer ce mot $caab$ et wv : 1) $w = \varepsilon$ et $v = caab$; 2) $w = c$ et $v = aab$; 3) $w = ca$ et $v = ab$. Les quasipériodes correspondantes de la forme $\mu_w(Pal(v))$ sont respectivement : 1) $Pal(caab) = cacacbcacac$; 2) $\mu_c(Pal(aab)) = cacacbcaca$; 3) $\mu_{ca}(Pal(ab)) = cacacbca$.

Remarque 5.1.10. Le théorème 5.1.9 montre en particulier que si un mot épisturmien $\mathbf{t}$ est dirigé par un mot spinné $\breve{\Delta}$ de spins ultimement L, alors $\mathbf{t}$ possède une infinité de quasipériodes puisque il y a une infinité de factorisations de $\breve{\Delta}$ dans la forme donnée (i.e., tout décalage positif d'un mot épistandard est quasipériodique).

Illustrons le théorème 5.1.9 par quelques exemples. D'abord nous présentons un exemple de mot épisturmien possédant une infinité de quasipériodes, mais qui n'est pas épistandard et dont les mots directeurs possèdent une infinité de lettres de spin L et de lettres de spin R (i.e. *wavy*).

Exemple 5.1.11. Considérons le mot épisturmien $\mathbf{t}$ de mot directeur normalisé $\breve{\Delta} = abc(\bar{b}a)^\omega$. D'après le théorème 3.0.1, on observe que $\mathbf{t}$ a une infinité de mots directeurs :

$$\breve{\Delta} = abc(\bar{b}a)^\omega \quad \text{et} \quad \breve{\Delta}_i = a\bar{b}\bar{c}(\bar{b}\bar{a})^i ba(\bar{b}a)^\omega \quad \text{pour tout } i \geq 0,$$

possédant tous une infinité de lettres de spin L et de lettres de spin R. Alors d'après le théorème 5.1.9, l'ensemble des quasipériodes de $\mathbf{t}$ consiste en $Pal(abc) = abacaba$ et, pour tout $i \geq 0$, les mots :

$$\mu_{\breve{w}}(bab) \quad \text{et} \quad \mu_{\breve{w}}(bab)a \quad \text{où } \breve{w} = a\bar{b}\bar{c}(\bar{b}\bar{a})^i.$$

Notons que $Pal(ba) = bab$ et $S_{\breve{w}}^{-1}Pal(w) = (Pal(w)a^{-1})^{-1}Pal(w) = a$.

Donnons maintenant quelques exemples de mots épisturmiens quasipériodiques ayant un nombre fini de quasipériodes.

Exemple 5.1.12. Considérons le mot épisturmien quasipériodique $\mu_{adbc\bar{d}}(a\mathbf{r})$ où $\mathbf{r}$ est le mot de Tribonacci ; il a seulement deux mots directeurs : $adbc\bar{d}(a\bar{b}\bar{c})^\omega$ et $ad\bar{b}\bar{c}d(a\bar{b}\bar{c})^\omega$ (le premier étant son mot directeur normalisé). Donc $\mathbf{t}$ possède *seulement* une quasipériode, à savoir $Pal(adbc) = adababadacadabada$. De même, le mot épisturmien $\mu_{cbaa}(c\mathbf{r})$, dirigé par exactement trois mots infinis spinnés différents $(cbaa(\bar{a}\bar{b}c)^\omega$, $cba\bar{a}a(\bar{b}c\bar{a})^\omega$, et $cb\bar{a}aa(\bar{b}c\bar{a})^\omega)$, a seulement deux quasipériodes : $Pal(cba) = cbcacbc$ et $Pal(cbaa) = cbcacbcacbc$.

On peut également construire des mots épisturmiens possédant exactement k quasipériodes pour un entier fixé quelconque $k \geq 1$:

Exemple 5.1.13. Considérons le mot épisturmien de mot directeur normalisé $\breve{\Delta} = abc\bar{a}(b\bar{c}\bar{a})^\omega$. D'après le théorème 3.0.1, $\mathbf{t}$ a exactement deux mots directeurs : $\breve{\Delta}$ et $\bar{a}\bar{b}\bar{c}a(b\bar{c}\bar{a})^\omega$. Donc, d'après le théorème 5.1.9, $\mathbf{t}$ a seulement une quasipériode, $Pal(abc)$. Maintenant, pour construire un mot épisturmien possédant exactement k quasipériodes pour un entier fixé $k \geq 1$, on considère $\breve{w}\breve{\Delta} = \breve{w}abc\bar{a}(b\bar{c}\bar{a})^\omega$ où $\breve{w}$ est une version spinnée de n'importe quel mot w de spin L avec $\mathrm{Alph}(w) \cap \{a, b, c\} = \emptyset$ et $|Pal(w)| = k - 1$. Alors, d'après le théorème 5.1.9, l'ensemble des quasipériodes du mot épisturmien dirigé par $\breve{w}\breve{\Delta}$ est constitué des k mots :

$$\mu_{\breve{w}}(Pal(abc))p \quad \text{où } p \text{ est un préfixe de } Pal(w).$$

Par exemple, $d^{k-1}\breve{\Delta} = d^{k-1}abc\bar{a}(b\bar{c}\bar{a})^\omega$ dirige un mot épisturmien avec exactement k quasipériodes : $\mu_d^{k-1}(Pal(abc))p = d^{k-1}ad^{k-1}bd^{k-1}ad^{k-1}cd^{k-1}ad^{k-1}bd^{k-1}ap$ où p est un préfixe de d^{k-1}.

Ainsi, le théorème 5.1.9 permet d'obtenir directement la caractérisation suivante des mots épisturmiens quasipériodiques :

Théorème 5.1.14. *Un mot épisturmien est quasipériodique si et seulement s'il existe un mot spinné $\breve{w}$, un mot v de spin L et une version spinnée $\breve{\mathbf{y}}$ d'un mot $\mathbf{y}$ de spin L tels que $\mathbf{t}$ est dirigé par $\breve{w}v\breve{\mathbf{y}}$ avec* $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$.

En utilisant les mots directeurs normalisés, on obtient le résultat suivant, plus facilement utilisable pour décider si un mot épisturmien donné est quasipériodique ou non :

Théorème 5.1.15.
Un mot épisturmien $\mathbf{t}$ est quasipériodique si et seulement si le mot directeur normalisé (unique) de $\mathbf{t}$ prend la forme $\breve{w}av_1\bar{a}v_2\bar{a}\cdots v_k\bar{a}v\breve{\mathbf{y}}$ pour une lettre a de spin L, un mot spinné $\breve{w}$, des mots de spin L sans a v, v_1, …, v_k ($k \geq 0$), et une version spinnée $\breve{\mathbf{y}}$ d'un mot $\mathbf{y}$ de spin L tels que $\mathrm{Alph}(av) = \mathrm{Alph}(av\mathbf{y})$.

Exemple 5.1.16. Les mots épisturmiens ayant pour mot directeur $(a\bar{b}\bar{c})^\omega$, $(a\bar{b}c\bar{a}b\bar{c})^\omega$ ou encore $(a\bar{b}\bar{a}ca\bar{a}\bar{b}cbc\bar{b})^\omega$ ne sont pas quasipériodiques alors que celui ayant pour mot directeur normalisé $ab\bar{a}bc\bar{c}(\bar{a}bc)^\omega$ est quasipériodique (il est aussi dirigé par $\bar{a}\bar{b}abc\bar{c}(\bar{a}bc)^\omega$).

Remarque 5.1.17. Une conséquence du théorème 5.1.14 est que tout mot épisturmien quasipériodique possède au moins deux mots directeurs. En effet, si $\mathbf{t}$ est un mot épisturmien quasipériodique, alors $\mathbf{t}$ a un mot directeur de la forme $\breve{\Delta} = \breve{w}v\breve{\mathbf{y}}$ où les mots $\breve{w}$ et $\breve{\mathbf{y}}$ sont des versions spinnées de mots de spin L w et $\mathbf{y}$, et v est un mot de spin L tel que $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$. Si $\breve{\mathbf{y}}$ (et donc $\breve{\Delta}$) a tous ses spins ultimement L, alors l'item 2 de la partie *iii*) du théorème 3.0.1 montre que $\mathbf{t}$ a également un mot directeur possédant une infinité de lettres de spin L et de spin R. Supposons maintenant que $\breve{\mathbf{y}}$ n'a pas ses spins ultimement L. Alors $\breve{\mathbf{y}}$ doit avoir une infinité de lettres de spin R. Choisissons x la lettre de spin R la plus à gauche dans $\breve{\mathbf{y}}$. Alors $\breve{\mathbf{y}}$ commence par $u\bar{x}$ pour un mot u de spin L (possiblement vide). Donc, puisque x apparaît dans v, $\breve{\Delta} = \breve{w}v\breve{\mathbf{y}}$ contient le facteur $vu\bar{x} = v'xv''u\bar{x}$. Ainsi $\breve{\Delta}$ ne satisfait pas les conditions du théorème 3.0.8 puisqu'il contient un facteur de $x\mathcal{A}^*\bar{x}$, et donc $\mathbf{t}$ n'a pas un mot directeur unique. De plus, on déduit facilement des théorèmes 5.1.14 et 3.0.8 que tout mot épisturmien ayant un unique mot directeur est nécessairement non-quasipériodique.

La remarque précédente pourrait laisser penser que tout mot épisturmien est non-quasipériodique si et seulement s'il possède un unique mot directeur, mais ce n'est pas le cas. Par exemple, $ba(\bar{b}c\bar{a})^\omega$ et $\bar{b}\bar{a}b(c\bar{a}\bar{b})^\omega$ dirigent tous deux le *même* mot épisturmien non-quasipériodique $\mathbf{t}$ d'après les théorèmes 3.0.1 et 5.1.14. Ces deux mot infinis spinnés sont les seuls mots directeurs de $\mathbf{t}$, ce qui pourrait mener à croire que tout mot épisturmien est non-quasipériodique si et seulement s'il possède un nombre fini de mots directeurs. Mais encore une fois, ce n'est pas le cas. Par exemple, comme établi dans l'exemple 5.1.12, $adbc\bar{d}(a\bar{b}\bar{c})^\omega$ et $ad\bar{b}\bar{c}d(a\bar{b}\bar{c})^\omega$ sont les deux seuls mots directeurs du mot épisturmien *quasipériodique* $\mu_{adbc\bar{d}}(a\mathbf{r})$ où $\mathbf{r}$ est le mot de Tribonacci. De plus il existe des mots épisturmiens non-quasipériodiques, comme celui dirigé par $(ab\bar{b}\bar{c})^\omega$, qui possèdent une infinité de mots directeurs (les mots $(ab\bar{b}\bar{c})^n a\bar{b}\bar{c}(ab\bar{b}\bar{c})^\omega$ sont des mots directeurs deux à deux distincts). Néanmoins, dans le cas sturmien, on montre qu'un mot sturmien est non-quasipériodique si et seulement s'il possède un mot directeur unique.

5.2 Quasipériodicité des morphismes épisturmiens

La quasipériodicité des mots épistandard peut être expliquée par la forte quasipériodicité des morphismes épistandard.

Théorème 5.2.1. *Soit v un mot sur un alphabet $\mathcal{A}$ contenant au moins deux lettres. Le morphisme épistandard μ_v est fortement quasipériodique si et seulement si* $\mathrm{Alph}(v) = \mathcal{A}$. *De plus $Pal(v)$ est une quasipériode de $\mu_v(\mathbf{w})$ pour tout mot infini $\mathbf{w}$.*

Remarque 5.2.2. Notons que tout mot infini Δ peut être écrit $\Delta = v\mathbf{y}$ avec $\mathrm{Alph}(v) = \mathrm{Alph}(v\mathbf{y})$. Donc si $\mathbf{t}$ (resp. $\mathbf{t}'$) est le mot épistandard dirigé par Δ (resp. $\mathbf{y}$), alors $\mathbf{t} = \mu_v(\mathbf{t}')$ et $\mathbf{t}$ est quasipériodique d'après le théorème 5.2.1. La même approche permet de montrer que si un mot épisturmien est dirigé par un mot infini spinné dont tous les spins sont ultimement L, alors il est quasipériodique.

À la lecture des théorèmes 5.2.1 et 5.1.9, on pourrait penser qu'un morphisme épisturmien $\mu_{\breve{u}}$ est fortement quasipériodique si et seulement si $\breve{u}$ possède un facteur v de spin L, mais ce n'est pas le cas. En effet, il existe des mots $\breve{u}$ de spin R tels que $\mu_{\breve{u}}$ est fortement quasipériodique, comme le montre le résultat suivant.

Lemme 5.2.3. *Soit $\breve{u}$ un mot fini spinné sur $\mathcal{A} \cup \bar{\mathcal{A}}$ et soit a une lettre de $\mathcal{A}$. S'il existe des mots v, y de spin L et un mot spinné $\breve{w}$ tels que $\breve{u} = \breve{w}\bar{a}\bar{v}\bar{y}$ et $\mathrm{Alph}(v) = \mathcal{A} \setminus \{a\}$, alors pour tout mot infini $\mathbf{t}$, $\mu_{\breve{u}}(a\mathbf{t})$ est quasipériodique.*

Par exemple, le mot $\bar{a}\bar{b}\bar{c}\bar{a}\bar{b}$ vérifie la condition du lemme précédent pour chacune des lettres a, b et c, donc $\mu_{\bar{a}\bar{b}\bar{c}\bar{a}\bar{b}}$ est fortement quasipériodique sur $\{a, b, c\}$.

La caractérisation des morphismes épisturmiens fortement quasipériodiques est en fait la suivante :

Théorème 5.2.4. *Soit $\mathcal{A}$ un alphabet contenant au moins trois lettres. Un morphisme épisturmien est fortement quasipériodique sur $\mathcal{A}$ si et seulement si son mot directeur normalisé $\breve{u}$ vérifie l'une des trois conditions :*

i) $\breve{u} = \breve{w}av_1\bar{a}v_2\bar{a}\cdots v_k\bar{a}v\breve{y}$
où a est une lettre de $\mathcal{A}$ (de spin L), $\breve{w}$, $\breve{y}$ sont des mots spinnés, $v_1, \ldots, v_k$ ($k \geq 0$) sont des mots de spin L sans a, et v est un mot de spin L tel que $\mathrm{Alph}(av) = \mathrm{Alph}(av\breve{y})$.

ii) Pour toute lettre a dans $\mathcal{A}$, $\breve{u} = \breve{w}\bar{a}\bar{v}\bar{y}$ pour un mot spinné $\breve{w}$ et des mots v, y de spin L tels que $\mathrm{Alph}(v) = \mathcal{A} \setminus \{a\}$.

iii) $\breve{u}$ vérifie le cas ii) pour toutes les lettres de $\mathcal{A}$ sauf pour une lettre $a \in \mathcal{A}$ telle que $\breve{u} = \breve{w}v\bar{a}\bar{y}$ où v et y sont des mots de spin L vérifiant $\mathrm{Alph}(v) = \mathcal{A} \setminus \{a\}$.

Chapitre 6

Conclusion

Je viens de présenter les résultats obtenus sur la quasipériodicité de toute une famille de mots et de morphismes : les mots et morphismes sturmiens et épisturmiens.

Afin d'avoir une compréhension plus générale des propriétés de quasipériodicité des morphismes, nous avons poursuivi ce travail d'étude de la quasipériodicité avec Gwénaël Richomme, cette fois dans le cadre de morphismes quelconques [LR2013].

Rappelons que les morphismes faiblement quasipériodiques et fortement quasipériodiques transforment respectivement au moins un mot ou tout mot non quasipériodique en mot quasipériodique. En considérant ces morphismes à la fois sur les mots finis et infinis, on obtient quatre familles de morphismes.

Nous avons montré que les (seules) relations entre ces différentes familles de morphismes sont celles de la figure 6.1.

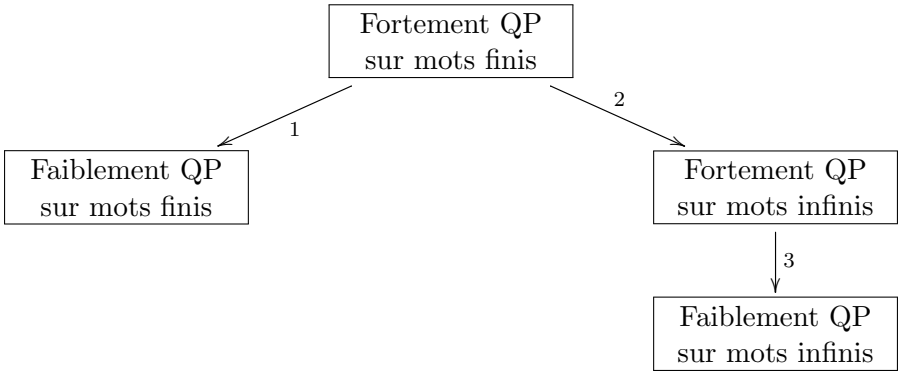


FIGURE 6.1 – Relations de base

Les relations 1 et 3 découlent des définitions, nous avons donc simplement prouvé la relation 2 et fourni des contre-exemples aux relations réciproques.

Les techniques développées pour l'étude de la quasipériodicité des morphismes sturmiens et épisturmiens n'étant plus utilisables dans ce cadre général, nous avons proposé un automate $\mathcal{A}_q(f)$ qui permet de reconnaître les mots u tels que pour un morphisme donné f et un mot donné q, $f(u)$ est q-quasipériodique ou égal à q ou au mot vide ε.

Soit $\mathcal{Q}(f)$ l'ensemble de tous les mots q tels que, pour toutes les lettres α de A, $|q| \leq 2|f(\alpha)|$ et q

est facteur de $f(\alpha)^3$. On peut montrer que $\mathcal{Q}(f)$ est l'ensemble de toutes les quasipériodes possibles d'un mot de la forme $f(u)$. Ceci implique la caractérisation suivante des morphismes fortement quasipériodiques sur les mots finis :

Proposition 6.0.1. *Un morphisme f est fortement quasipériodique sur les mots finis si et seulement si, pour toute lettre α, le mot $f(\alpha)$ est quasipériodique, et*

$$A^* = \bigcup_{q \in \mathcal{Q}(f)} \mathcal{L}(\mathcal{A}_q)$$

où $\mathcal{L}(\mathcal{A}_q)$ est le langage reconnu par l'automate $\mathcal{A}_q$.

Dans le cas des mots infinis, nous avons obtenu un résultat similaire en adaptant l'automate utilisé pour les mots finis, et en remarquant qu'un mot infini était q-quasipériodique si et seulement si tous ses préfixes étaient reconnus par cet automate.

Nous avons enfin proposé quelques pistes pour déterminer si un morphisme est faiblement quasipériodique, cependant sans obtenir de caractérisation complète (voir [LR2013]).

Nous allons maintenant aborder la deuxième partie de ce mémoire, qui porte toujours sur la structure, mais cette fois en analyse musicale.

Deuxième partie

Contribution à l'analyse musicale par ordinateur

Chapitre 7

Introduction à la partie

Comme mentionné dans l'introduction générale, la motivation de cette recherche est la même que dans la partie précédente : il s'agit de mieux comprendre la structure des objets sur lesquels on travaille, ici des œuvres musicales. Bien que la combinatoire des mots puisse être utilisée pour modéliser certains aspects de la musique, notamment les rythmes ou les suites d'accords [2, 9, 19], les méthodes que nous utiliserons dans cette partie sont plutôt issues de l'algorithmique du texte. De même que pour étudier la quasipériodicité des mots, on va chercher les motifs qui apparaissent plusieurs fois, se répètent à la suite, se chevauchent. Dans une fugue par exemple, le thème principal, appelé sujet, est d'abord introduit par une voix seule, puis repris successivement par toutes les autres voix. L'entrée d'une voix se fait en général à la fin de l'exposition du sujet par la voix précédente, mais il peut parfois y avoir de légers chevauchements. Ainsi, d'une certaine manière, une fugue n'est qu'une partition dont certains segments sont quasipériodiques. Seulement les méthodes de recherche de la structure ne peuvent pas être les mêmes ici que dans le cas des mots sturmiens et épisturmiens. La première raison est que la musique du répertoire classique, sur laquelle nous focalisons notre étude, n'est pas engendrée de façon systématique. Il n'existe pas de morphisme "fuguien", par exemple... bien que les morphismes puissent être utilisés pour composer de la musique [1]. La deuxième raison est que lorsqu'un motif se répète en musique, il est rarement repris de façon exactement identique : il est donc nécessaire d'utiliser des méthodes de recherche approchée au lieu de rechercher uniquement les répétitions exactes.

Toutes les études réalisées dans cette partie ont le même objectif en filigrane : proposer une *automatisation de l'analyse de la structure*, et à plus long terme de la forme musicale.

On cherche donc à obtenir un schéma d'une pièce présentant les éléments saillants la composant et sa forme générale, comme dans l'exemple de la figure 7.1 issu de l'analyse des fugues de Jean-Sébastien Bach par Siglind Bruhn [13].

De mon point de vue, l'analyse automatique de la structure, tout comme l'analyse manuelle, doit nécessairement se fonder sur plusieurs critères. On peut s'appuyer sur la connaissance des motifs et de leurs répétitions, aussi bien d'un point de vue "mélodique" (ou disons de la hauteur des notes) que rythmique, sur la détection des phrases musicales, mais aussi sur la détection des cassures dans la régularité, qui peuvent se manifester par des modulations (des marches harmoniques), des cadences, du changement de texture musicale. En résumé, c'est par la détection de tous les éléments compositionnels pris individuellement ou conjointement, que l'on peut trouver la structure "multi-échelle" de la partition, aussi bien au niveau local que par la compréhension de son organisation globale.

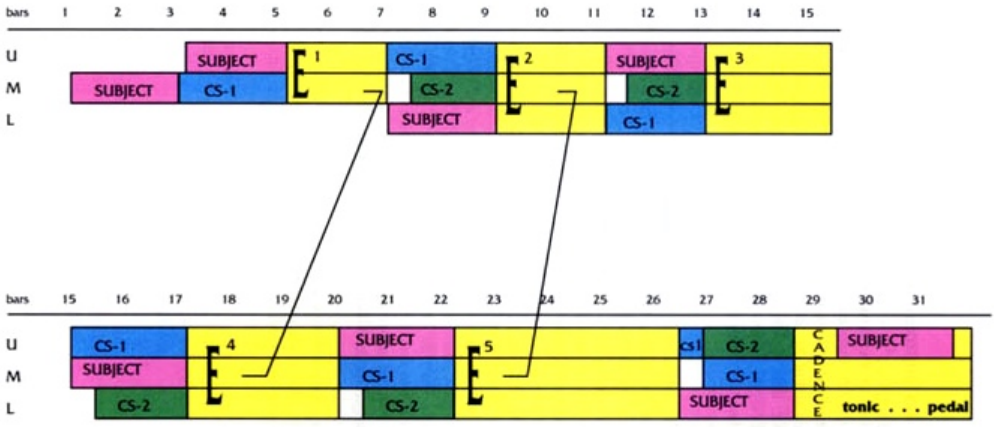


FIGURE 7.1 – Schéma de la fugue n° 2 de J.-S. Bach par Siglind Bruhn.

Ce travail ne peut se faire sans l'analyse experte des analystes "humains" de la musique, c'est pourquoi chaque étude a été réalisée en se basant sur des précis musicologiques, ou les expertises de musicologues avec qui nous collaborons.

Dans cette partie, nous allons d'abord présenter nos travaux sur l'analyse de la fugue, ainsi que quelques propositions pour l'analyse de la forme sonate, dans le cas où l'on connaît les différentes voix d'une partition. Puis nous aborderons l'analyse de partitions polyphoniques où les voix ne sont pas séparées, en proposant des méthodes prenant en compte le rythme ou la texture de la partition. Nous terminerons par nos perspectives à court et moyen terme pour améliorer l'analyse de la structure et de la forme dans le cas général.

Chapitre 8

Formes musicales : fugues, forme sonate.

L'analyse musicale consiste en l'étude des différents éléments constituant une œuvre. Elle est un outil indispensable pour les jeunes musiciens afin d'apprendre à composer de la musique, et permet souvent de mieux comprendre et de mieux interpréter une œuvre musicale. Il s'agit en général de repérer entre autres les lignes mélodiques, les répétitions (motifs), les accords formés par les notes jouées simultanément (harmonie), leur agencement entre les différentes voix et leur densité (texture) et à un niveau plus global, de repérer la structure d'un passage ou de l'œuvre toute entière. Nous nous sommes d'abord intéressés à l'analyse de la fugue, une forme majeure de la musique occidentale classique, souvent utilisée dans les cours d'analyse musicale et de composition.

8.1 L'analyse de la forme musicale

Les problématiques liées à l'analyse automatique de la structure et de la forme musicale ont fait l'objet d'un chapitre de livre écrit en collaboration avec Mathieu Giraud et Richard Groult, et édité par David Meredith [GGL2016].

Les formes musicales décrivent la façon dont une œuvre musicale est structurée, expliquant comment les parties interagissent au travers des répétitions, des contrastes et des variations : la répétition apporte l'unité, et les variations suscitent l'intérêt. L'étude des formes musicales, fondamentale dans l'éducation musicale, permet entre autres une meilleure connaissance des règles de composition, et constitue une première approche essentielle à une bonne interprétation d'un morceau. Bien sûr, toutes les pièces musicales ne peuvent pas être cantonnées à un schéma restrictif formel. Cependant, les formes existent et sont un aspect important de ce que l'auditeur attend en écoutant de la musique [20]. La formalisation de nombreuses formes musicales a été faite dans des ouvrages du XIXème siècle afin d'apprendre la composition aux jeunes musiciens [87, 23, 63]. Depuis, de nombreuses théories sur les formes musicales et le moyen de les analyser ont émergé [72, 85, 97, 86]. Les méthodes systématiques pour l'analyse de la musique incluent en général une certaine formalisation de la structure à large échelle, comme par exemple l'analyse Schenkerienne [96], la théorie de LaRue [55], ou la théorie générative de la musique tonale [57].

La recherche sur les théories des formes musicales est toujours à l'heure actuelle un domaine très actif de la musicologie [17, 18].

Le tableau 8.1 liste les formes les plus courantes, du point de vue de la *segmentation* : les formes sont données ici comme une succession de parties dénommées par des étiquettes.

Fugue	Forme contrapuntique	
Variations	A A' A" A"'	
Binaire	AA', AB, AAB, ABB, AA'BB'	
Ternaire	ABA, ABA'	
Rondo	ABACADA...	où A : 'refrain', B, C, D : 'couplet'
Forme Sonate	AB-Dev-AB	où A : thème/zone 'principal(e)'
Forme Rondo-Sonate	AB-AC-AB	et B : thème/zone 'secondaire'
Forme cyclique		
Formes libres		

TABLE 8.1 – Formes musicales d'un point de vue thématique. Les lettres désignent les sections contrastantes.

Ainsi, la forme binaire est constituée de deux parties, l'une ou les deux pouvant être répétées, souvent avec des variations. Ces parties peuvent utiliser le même matériel thématique ou un matériel différent. En général, elles sont de longueur comparable, mais la deuxième partie peut également être plus longue.

Dans les formes ternaires, une partie contrastante s'insère entre les deux répétitions de la première partie. Une forme ternaire peut être simple (aria da capo ou forme lied), ou composée de formes binaires (forme trio, comme le Menuet/Scherzo), voire de formes ternaires. Les formes sonates incluent une exposition, un développement, et une réexposition, généralement à partir de deux zones thématiques.

Cette segmentation ne présente qu'un aspect de la forme. Une compréhension correcte d'une forme ternaire nécessite d'expliquer de quelle manière (tonalité, texture, rythme...) la section contrastante diffère de la partie initiale. De même, plusieurs formes incluent une tension à haut-niveau et une dynamique à expliciter. La forme sonate combine ces principes, incluant un chemin tonal : la pièce part de la tonique pour aller dans une clé secondaire, souvent la dominante, puis revenir à la tonique.

Bien sûr, il y a aussi des formes plus *libres*, plus difficiles à classifier selon une vue basée sur la segmentation. Les préludes et les inventions de la période baroque sont souvent vus comme des formes libres. Cependant, cela ne veut pas dire qu'ils ne sont pas structurés, mais plutôt que leur organisation dépend d'autres composants, tels que la texture ou la tension. Un cas extrême est celui de certaines formes *ouvertes*, où l'ordre d'exécution des sections n'est pas fixé par le compositeur, mais peut tout de même inclure une conception formelle de certains composants.

Dans tous les cas, une analyse musicale complète devrait révéler tout principe organisationnel, au delà d'une simple segmentation, permettant de mieux comprendre la forme musicale [11]. Cette analyse contient plusieurs enjeux : non seulement une segmentation correcte nécessite d'identifier des phrases, des frontières et des similarités entre parties, mais il faut également pouvoir proposer une sémantique pour cette segmentation, indiquant la fonction formelle des sections, la façon dont elles sont liées, ainsi qu'un schéma global.

Dans ce qui suit, nous présentons nos travaux sur l'analyse de la structure de la fugue, avant de proposer des idées pour amener à l'analyse d'une forme plus complexe : la forme sonate.

8.2 Analyse automatique de fugues

Les résultats présentés dans ce chapitre ont fait l'objet de 2 publications dans des conférences internationales [GGL2012a, GGL2012b], et d'un article dans un journal de référence en informatique musicale (Computer Music Journal) [GGLL2015].

La fugue est l'une des formes les plus notables de la musique classique occidentale. De nombreux compositeurs ont écrit des fugues ou ont inclus des parties fuguées dans de grandes oeuvres.

C'est pourquoi cette forme est beaucoup étudiée dans les classes d'analyse et de composition des conservatoires et des écoles de musique [5]. De nombreux traités ont été écrits sur les fugues et le contrepoint en général, comme [31], [32], [74] ou [52]. Certaines études comprennent une analyse complète du Clavier Bien Tempéré de Jean-Sébastien Bach, qui forme un corpus particulièrement consistant explorant les 24 tonalités majeures et mineures au travers de 48 préludes et fugues ([81] et [13]).

L'origine du mot fugue viendrait des mots latins *fugere* et *fugare*, signifiant respectivement *fuir* et *poursuivre*, qui représentent bien le jeu des différentes voix reprenant successivement les mêmes motifs musicaux. En effet, une fugue est une pièce polyphonique construite sur un ou plusieurs thèmes musicaux : un *sujet* et, souvent, un ou plusieurs *contre-sujets*, ainsi que d'autres motifs et des sections modulantes appelées épisodes (voir la figure 8.1). Les analyses musicales sont construites sur ces motifs et sections. En général, le sujet est d'abord exposé à une voix, puis repris à une autre voix pendant que la première voix joue éventuellement une première fois un contre-sujet, et d'autres occurrences se suivent ainsi sur toutes les voix, alternant avec d'autres motifs ou des sections modulantes appelées *épisodes*. Les motifs ainsi répétés peuvent l'être dans leur forme initiale, ou plus souvent, modifiés ou transposés, construisant ainsi une texture harmonique complexe. Les analyses musicales des fugues sont en général basées sur la reconnaissance de ces motifs et des différentes sections où ils apparaissent (exposition, épisodes, coda).

Nous avons proposé différents algorithmes pour réaliser l'analyse automatique d'une fugue, à partir de fichiers où les voix (les différentes lignes musicales) étaient séparées (il s'agissait de fichiers Humdrum -.krn). Ces algorithmes se basent sur les similarités musicales diatoniques (sans se préoccuper des altérations, dièses ou bémols) des intervalles entre les notes successives, et nous avons pu détecter les sujets, contre-sujets, ainsi que les suites harmoniques à l'intérieur des épisodes. Nous avons également fourni des outils pour détecter les degrés, les cadences simples et les pédales, qui sont des éléments importants de l'analyse harmonique d'une oeuvre, et proposé une segmentation de la fugue en exposition et partie épisodiques.

Notre corpus expérimental a été constitué de 36 fugues de Bach (1er livre du Clavier Bien Tempéré) et de Shostakovitch. Afin de valider nos algorithmes, nous avons réalisé des fichiers d'analyse de référence, formalisés grâce à plusieurs analyses musicologiques de la littérature (en particulier les ouvrages de Siglind Brühn pour les fugues de Bach [13]), et nous les avons mis à la disposition de la communauté. Les résultats nous ont permis de fournir une ébauche de la structure des fugues testées. Dans cette partie, nous détaillons les méthodes que nous avons développées pour l'analyse de la fugue et les résultats obtenus.

8.2.1 Représentation de la musique

Une *note* x est représentée par un triplet (p, o, d), où p est la hauteur de la note (*pitch* en anglais), o l'instant de début de la note (*onset* en anglais), et d sa durée. La hauteur est un entier pris dans une

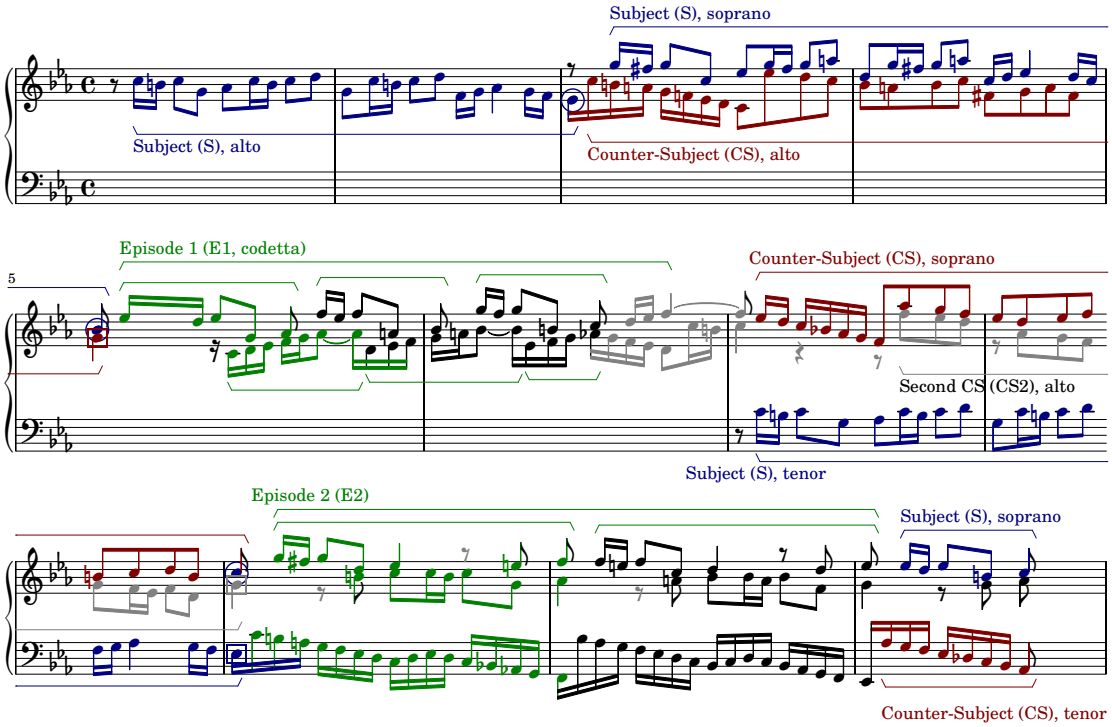
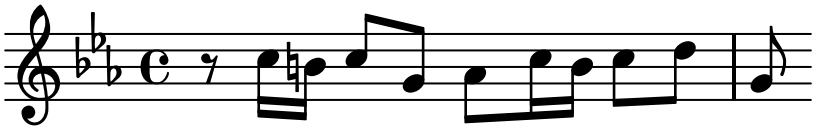


FIGURE 8.1 – Premières mesures de la fugue n° 2 de J.-S. Bach's en Do mineur (BWV 847). Les sujets (S), contre-sujets (CS1 et CS2), et les deux premiers épisodes (E1 et E2). Les notes terminant les sujets, correspondant à une analyse de référence, sont entourées, et les notes terminant les contre-sujets sont encadrées. À l'intérieur des épisodes, les accolades montrent les motifs récurrents (trois occurrences dans E1, et deux occurrences dans E2).

échelle de valeur pouvant représenter une information diatonique (basée sur la succession des notes de la gamme, ce qui revient à ne pas considérer les altérations) ou par demi-tons. On considère des *suites de notes* ordonnées $x_1 \dots x_m$, c'est-à-dire $x_1 = (p_1, o_1, d_1), \dots, x_m = (p_m, o_m, d_m)$, où $0 \leq o_1 \leq o_2 \leq \dots \leq o_m$ (voir la figure 8.2). Une suite est *monophonique* si deux notes ne sonnent jamais ensemble, c'est-à-dire si pour tout i avec $1 \leq i < m$, on a $o_i + d_i \leq o_{i+1}$. Il y a un *silence* entre deux notes x_i et x_{i+1} si $o_i + d_i < o_{i+1}$, et la durée de ce silence est $o_{i+1} - (o_i + d_i)$. Pour être capable de reconnaître la similarité entre des suites transposées (i.e. jouées à des hauteurs différentes), on considère les hauteurs de notes relatives, c'est-à-dire les *intervalles* de hauteur. Une suite d'intervalles est définie par ${}^{\Delta}x_2 \dots {}^{\Delta}x_m$, où ${}^{\Delta}x_i = ({}^{\Delta}p_i, o_i, d_i)$ et ${}^{\Delta}p_i = p_i - p_{i-1}$.

8.2.2 Une adaptation de l'algorithme de Mongeau-Sankoff

Le score de similarité entre un motif et le reste de la fugue peut être calculé par programmation dynamique avec des équations semblables à celles de [68]. Soit $x = x_1 \dots x_m$ un motif et $y = y_1 \dots y_n$ une voix. Pour $a \leq m$, soit $S(a, b)$ le meilleur nombre d'intervalles coïncidant (on parle de *match*) en alignant le début du motif $x_1 \dots x_a$ avec $y_i \dots y_b$, une partie de la voix y terminant à b, et soit $S_f(m, b)$ le meilleur nombre d'intervalles coïncidant en alignant le motif complet $x_1 \dots x_m$ (un *candidat*) contre la même partie $y_i \dots y_b$. Les tables S et S_f sont calculées par programmation



pitch p	72	71	72	67	68	72	71	72	74	67
interval Δp		−1	1	−5	1	4	−1	1	2	−7
onset o	2	3	4	6	8	10	11	12	14	16
duration d	1	1	2	2	2	1	1	2	2	2

FIGURE 8.2 – Une suite monophonique de notes (début de la fugue n° 2 de J.-S. Bach) representée par des triplets (p, o, d) ou $({}^{\Delta}p, o, d)$. Dans cet exemple, les "onsets" et les durées sont comptées en doubles croches, et les hauteurs et intervalles sont comptés en demis tons avec les valeurs MIDI standard.

dynamique en utilisant les équations suivantes :

$$\begin{cases} S(1,b) = & 0 \\ \forall a \geq 2, S(a,b) = & S(a-1,b-1) + \delta({}^{\Delta}x_a, {}^{\Delta}y_b) \qquad \text{(match, substitution)} \\ \forall m \geq 2, S_f(m,b) = & S(m-1,b-1) + \delta_f({}^{\Delta}x_m, {}^{\Delta}y_b) \qquad \text{(terminaison)} \end{cases} \tag{8.1}$$

L'alignement peut être retrouvé en parcourant en arrière la table de programmation dynamique. Les fonctions de substitution δ et δ_f sont définies de la façon suivante (la relation $\approx$ est une relation de similarité sur les intervalles de hauteur) :

$$\delta(({}^{\Delta}p, o, d), ({}^{\Delta}p', o', d')) = \begin{cases} +1 & \text{si} \quad {}^{\Delta}p \approx {}^{\Delta}p' \quad \text{et} \quad d = d' \\ 0 & \text{si} \quad {}^{\Delta}p \not\approx {}^{\Delta}p' \quad \text{et} \quad d = d' \\ -\infty & \text{sinon } (d \neq d') \end{cases} \tag{8.2}$$

$$\delta_f(({}^{\Delta}p, o, d), ({}^{\Delta}p', o', d')) = \begin{cases} +1 & \text{si} \quad {}^{\Delta}p \approx {}^{\Delta}p' \\ 0 & \text{sinon } ({}^{\Delta}p \not\approx {}^{\Delta}p') \end{cases} \tag{8.3}$$

Notons que δ teste les intervalles de hauteur et les durées, alors que δ_f considère seulement les intervalles de hauteur. Les durées des premières notes (x_1 et y_1) ne sont pas testées, car en réalité l'algorithme compare ${}^{\Delta}x_2 \ldots {}^{\Delta}x_a$ et ${}^{\Delta}y_2 \ldots {}^{\Delta}y_b$. La raison est que la première et la dernière note des motifs thématiques sont plus fréquemment transformées que les autres notes, comme dans la figure 8.3.

Ces fonctions de substitution sont très conservatrices, dans le sens où elles autorisent les erreurs de substitution et l'égalité sur une longueur stricte, afin de trouver aussi peu de faux positifs que possible tout en conservant un taux de reconnaissance élevé. Par exemple, dans l'alignement de la figure 8.3, le motif à la mesure 2 coïncide parfaitement avec sa transposition diatonique T : $S_f(m2, T) = 5$. Cependant, dès que le motif est muté, le motif n'est plus reconnu : ici $S_f(m2, m8) = S_f(m2, m5) = -\infty$. Les équations peuvent être étendues pour considérer d'autres opérations d'édition. Par exemple, autoriser les opérations d'insertion et de suppression avec un

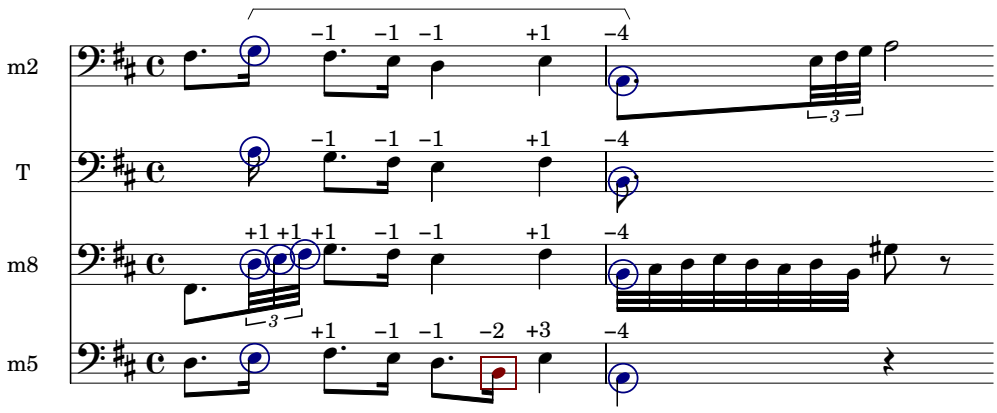


FIGURE 8.3 – Alignement de 4 motifs semblables dans la fugue n° 5 de J.-S. Bach. En considérant des pas diatoniques, les intervalles de hauteur du premier motif (m2) sont $({}^{\Delta}x_2..{}^{\Delta}x_6) = (-1,-1,-1,+1,-4)$. La ligne T présente une transposition (diatonique), non présente dans la fugue, qui donnerait une correspondance parfaite. Dans les occurrences réelles du motif, la première note (entourée) est soit fragmentée (m8) ou substituée (m5), et la longueur de la dernière note (entourée) n'est pas préservée. Le motif est modifié plus loin par une insertion (le si encadré) qui peut aussi être vue comme une fragmentation du ré noire).

score 0 donnerait $S_f(m2, m8) = 4$ et $S_f(m2, m5) = 2$. Des opérations de fragmentation appropriées avec un score +1 élèverait le score $S_f(m2, m5)$ à 4. La comparaison des valeurs S_f avec un seuil permet de décider si une occurrence est conservée ou non. Un seuil de 3 permet ici jusqu'à deux erreurs d'intervalles. Cependant, quand les contraintes sont relâchées, puisque quasiment tout le contenu d'une fugue est d'une certaine façon dérivé d'un sujet ou d'un contre-sujet, n'importe quelle partie pourrait s'aligner avec une partie du sujet ou un autre motif de base avec un seuil donné.

Plusieurs relations de similarité $\approx$ peuvent être choisies, de l'égalité stricte de hauteur à des classes définissant le contour mélodique ("+haut, -haut") [33]. Des classes d'intervalles intermédiaires peuvent être définies, comme les "pas/sauts" (*step/leap*) [16] ou les "intervalles de chevauchement partiel quantisés" (QPI) [56]. Le modèle QPI définit des intervalles *courts* (de un à trois demi-tons), *moyens* (de trois à 7 demi-tons), et *grand* (à partir de 6 demi-tons). Quand les noms des notes sont disponibles – ou quand ils peuvent être calculés par des algorithmes dédiés, tels que [66] – la similarité peut être directement calculée sur les hauteurs *diatoniques*, ce modèle étant tout à fait adapté à la musique tonale [76, 14, 41, 79].

8.2.3 Algorithmes pour l'analyse de fugue

Nous avons proposé plusieurs outils permettant de donner des informations structurelles sur les fugues. En utilisant l'adaptation de l'algorithme de Mongeau-Sankoff présentée ci-dessus et d'autres techniques discrètes, nous proposons de calculer la fin du sujet, puis d'inférer un contre-sujet (CS), voire un deuxième contre-sujet (CS2). Trouver ces motifs et la liste complète de leurs occurrences (y-compris celles dont la durée est augmentée ou celles qui sont inversées) ainsi que leur degré (dans la gamme) permet de former la structure de la fugue. Nous identifions les marches harmoniques, les cadences (simples) et les pédales. Nous combinons également certains de ces éléments pour obtenir une segmentation globale grâce à une modélisation statistique. Le résultat de ces algorithmes sur la

fugue n° 2 de J.-S. Bach est décrit dans la section “analyse calculée” (“Computed analysis”) au bas de la figure 8.5. Le résultat de notre méthode est donc constitué à la fois des différents éléments et de la prédiction de la segmentation globale.

Identification du sujet

Trouver la fin du sujet est nécessaire pour trouver le contre-sujet et construire la structure complète de la fugue, incluant toutes les occurrences de ces motifs. Comme nous l'avons déjà expliqué, au début de la fugue, le sujet est (en général) d'abord joué seul par une voix unique jusqu'à ce que la deuxième voix entre. La fugue n° 6 de l'opus 87 de Shostakovitch est un contre-exemple où la première entrée du sujet est doublée à l'octave par la voix de basse ; cela peut aussi être vu comme un changement de texture (voir la partie 9.2).

Cependant, dans la plupart des cas, la fin du sujet n'est pas située exactement au début de la deuxième voix. Par exemple, dans la fugue n° 2 de J.-S. Bach, si l'on se réfère aux quatre analyses de référence musicologiques que nous avons formalisées, le premier sujet termine sur la note mi bémol à l'alto, qui est située deux notes (deux doubles croches) avant le début de la voix de soprano (i.e., la première note entourée sur la figure 8.1). Plusieurs paramètres contribuent à cette perception. Métriquement, la phrase se termine sur un temps fort ; harmoniquement, les 5 notes précédentes (fa sol la bémol sol fa) suggèrent un accord de septième de dominante, qui se résoud sur le mi bémol, suggérant la tonique do mineur. Le sujet se termine par une succession de doubles croches avec de petits intervalles, alors que la note suivante (do) appartient au premier contre-sujet, séparée du mi bémol par un large saut et débutant une gamme descendante en doubles croches.

Dans les 36 fugues de notre corpus, la fin du sujet se trouve entre −8 et +6 notes (inclus) par rapport au début du contre-sujet (sauf pour la fugue n° 6 de l'opus 87 de Shostakovitch, −18 notes), et dans la majorité des cas, entre −4 et +1 notes (inclus, voir e_S dans la figure 8.4).

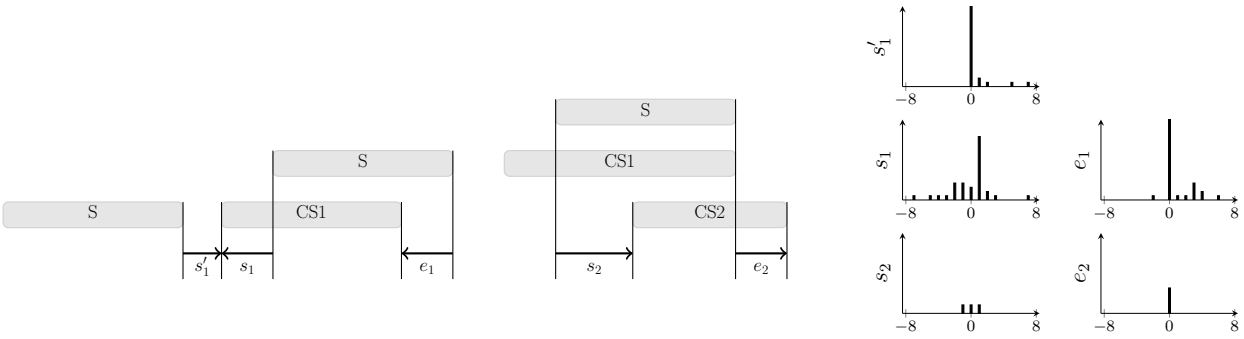


FIGURE 8.4 – Positions de début relatif (s, s') et de fin (e) des contre-sujets (1 : CS1, 2 : CS2) décrits dans l'analyse de référence. Ces positions sont calculées en nombre de notes par rapport aux positions de début et de fin du sujet correpondant (S). Les histogrammes montrent la distribution de ces valeurs sur les 36 fugues de notre corpus. En général $s'_1 = 0$ (CS1 commence juste après le sujet) et $e_1 = e_2 = 0$ (CS1 et CS2 terminent exactement à la fin du sujet).

Ici un algorithme simple, basé sur des similarités, est capable de détecter précisément la plupart des fins de sujet [GGL2012a]. À la première voix, on teste les motifs qui terminent entre 8 notes avant et 6 notes après le début de la deuxième voix : soit z la position de la note de la première voix jouée avant ou au début de la seconde voix, et $g \in [-8, 6]$. Chacun des motifs candidats,

terminant à $z + g$, est aligné sur toutes les voix. Cet alignement est effectué grâce à la fonction décrite dans la section précédente. Le candidat obtenant le meilleur score sur toutes ses occurrences est sélectionné : le sujet choisi est celui finissant à $z + g$ qui maximise la somme des scores de ses occurrences, c'est-à-dire la somme des valeurs $S_f(z + g, i)$ pour une position donnée i telle que $S_f(z + g, i)$ est plus grand ou égal à un seuil dépendant de la longueur du motif.

Identification du contre-sujet

Le premier contre-sujet (CS1) commence en général juste après le sujet (voir s'_1 dans la figure 8.4), sauf dans certains cas comme la fugue n° 16 de J.-S. Bach ou la fugue n° 2 de Shostakovitch. Le deuxième contre-sujet, lorsqu'il existe, commence plus tard, approximativement au début de la troisième entrée du sujet (voire après une codetta, c'est-à-dire un bref passage épisodique à l'intérieur de l'exposition), et termine approximativement à la fin de ce troisième sujet (et à la deuxième entrée du CS1). La figure 8.4 montre la distribution de ces positions relatives dans les pièces du corpus. Dans la majorité des cas, les contre-sujets terminent exactement à la même position que les sujets (voir les notes encadrées dans la figure 8.1), ce qui marque une transition entre l'exposition et la partie épisodique. Pour trouver CS1, on teste les motifs commençant juste après le sujet et terminant à la fin du second sujet environ (entre 2 notes avant et 4 notes après). On choisit le meilleur candidat, à nouveau en choisisssant le z qui maximise la somme des $S_f(z + g, i)$ pour toutes les occurrences i du motif candidat. Notons qu'une mauvaise détection du sujet peut mener à une mauvaise détection du premier contre-sujet. Le deuxième contre-sujet est encore plus difficile à trouver, car il peut être plus court, et sa position de début peut être difficile à prédire. De plus, les musicologues ne se mettent pas toujours d'accord sur la présence d'un deuxième contre-sujet.

Détection de toutes les occurrences

Une fois que les motifs S/CS1/CS2 ont été identifiés, il est possible de relancer l'algorithme en modifiant ses paramètres pour détecter plus d'occurrences de ces motifs. Les fonctions de substitution très conservatrices δ et δ_f peuvent être assouplies pour permettre la substitution de durée ou, au moins, les fragmentations et consolidations préservant la durée totale. Quand on abaisse le seuil global, on trouve des occurrences incomplètes de S qui ne sont pas des entrées convenables. Ces contraintes relâchées ne doivent pas être utilisées dans la phase d'identification des motifs car ils sont insuffisants pour détecter les fins. On regarde spécifiquement les "têtes" des sujets en cherchant les motifs se limitant au 5 premières notes du sujet (voir la figure 8.5).

On ajoute d'autres contraintes pour la détection de CS1/CS2, autorisant un CS1 seulement quand il y a un sujet en même temps et un CS2 seulement quand il y a un CS1 en position compatible. Finalement, dans certaines fugues, les sujets apparaissent inversés (i.e. les intervalles sont inversés) ou augmentés (i.e. les durées sont doublées). Quand on connaît le sujet, le même algorithme d'alignement trouve les inversions et les augmentations du sujet.

Degrés et tonalités locales

Après une exposition à la tonique (degré I) puis à la dominante (degré V), la fugue module souvent vers des tonalités relatives ou plus distantes, revenant à la tonique pour la conclusion. Déterminer ces tonalités locales fait partie intégrante d'une analyse complète. Ces tonalités pourraient être estimées par des algorithmes classiques comme dans les travaux [54, 100] et leurs extensions

ou les résultats plus récents de [60] ou [92]. Cependant, ces algorithmes pourraient assigner des tonalités incorrectes dans les passages très modulants comme les épisodes.

Nous n'utilisons pas de détection de clé mais une méthode plus simple, tirant partie de la détection du sujet et ne cherchant pas à étudier la tonalité des épisodes. On se concentre sur les sujets (qui sont les endroits où la tonalité locale est la plus stable) et on détermine l'intervalle de transposition entre le sujet courant et le premier sujet, qui est à la tonique (I). Pour cela, on calcule la hauteur moyenne, en demi-tons, de chaque entrée du sujet. Les *degrés* des sujets sont ensuite calculés relativement à la tonalité globale établie d'après le premier sujet. Quand le sujet est transposé à la dominante, de légères modifications peuvent apparaître (*réponse tonale*) en général dans les toutes premières notes du sujet pour amener un retour à la tonique. Ces modifications sont compensées par les autres notes à la hauteur moyenne, par conséquent le calcul des degrés reste correct. Cependant, ceci ne permet pas la détection des changements de tonalité à l'intérieur d'un sujet. Plusieurs sujets commencent en effet à la tonique et modulent vers la dominante. Néanmoins, connaître les degrés de la partie principale des sujets donne une vision du chemin tonal de la pièce complète. Par exemple, le chemin tonal de la fugue n° 2 de J.-S. Bach peut être vu dans la figure 8.5. Les degrés sont notés en chiffres romains au bas des diagrammes. Alors que l'exposition initiale et la fin de la fugue alternent entre la tonique (I) et soit la sous-dominante (IV), soit la dominante (V), les partie médianes atteignent la tonalité relative (III ou VI).

Marches harmoniques

Dans les fugues, les marches harmoniques se trouvent presque exclusivement dans les épisodes. Les détecter aide donc à trouver la structure de la fugue. Dans chaque voix, on cherche les motifs répétés qui se suivent, c'est-à-dire deux occurrences consécutives pat_1 et pat_2 du même motif, et éventuellement trois (pat_1, pat_2, pat_3). On utilise la fonction de substitution décrite dans la section précédente, qui nécessite que $S(pat_1, pat_i)$ soit au-dessus d'un certain seuil pour chaque occurrence pat_i.

On essaie de détecter les marches harmoniques partielles commençant à toutes les notes situées à chaque temps de la partition (ou immédiatement après).

On considère qu'il y a une suite lorsque des motifs sont répétés consécutivement dans au moins deux voix différentes avec la même transposition [GGL2012b]. Plusieurs périodes (longueurs de motifs) sont testées : un et deux temps (dépendant de la métrique), une et deux mesures. Lorsque tous les temps sont testés, la procédure ci-dessus trouve plusieurs suites chevauchantes, avec la même période, dès qu'il y a strictement plus de deux occurrences. Ces suites chevauchantes sont fusionnées en une suite unique.

Cette méthode trouve correctement la plupart des marches harmoniques. Au début de la fugue n° 2 de J.-S. Bach, (voir la figure 8.5), les deux marches harmoniques sont détectées, cependant la première, pendant la codetta, est raccourcie à cause du début décalé des deux voix. Les marches harmoniques sont de bons marqueurs des épisodes. Cependant, une situation courante est que la marche harmonique commence à la fin d'un motif thématique, continuant dans l'épisode (voir la figure 8.6).

Cadences et pédales

Les cadences et les pédales marquent les transitions structurelles de la fugue (fin de l'exposition, fin des épisodes, et coda avec une pédale). Sauf pour certains cas particuliers (voir la figure 8.8), les

```
== S [length 2 start +1/8]   == CS1 [length 2 start +1/16]       == cad
S  3, 11, 20, 29.5           S  7 [start -1/4], 15 [start 0]    * 17 (v:PAC)
A  1, 15                     A  3, 20                           * 22 (i:rIAC)
T  7, 26.5                   T  11                              * 28.5 (i:IAC)
                             SA 26.5 [change +1/2 end +1/8]     * 29.5 (I:PAC)
```

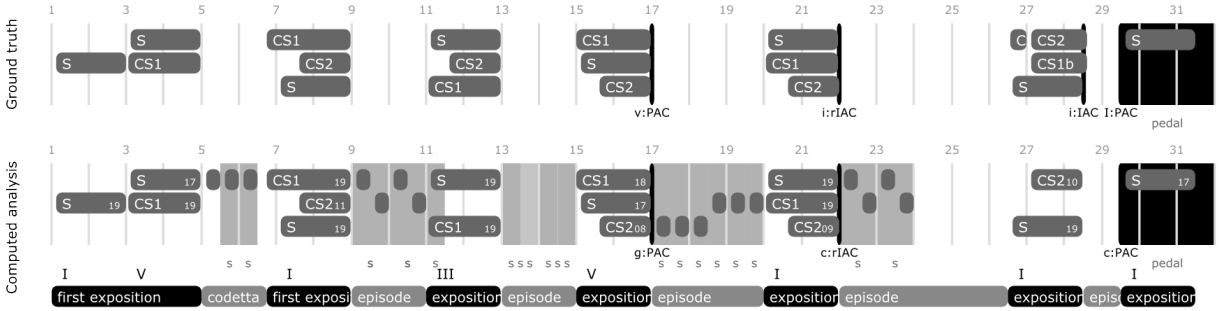


FIGURE 8.5 – Analyse de la Fugue n° 2 en Do mineur dans le 1er livre du Clavier Bien Tempéré de J.-S. Bach (BWV 847). (Top) Extrait du fichier des analyses de référence montrant les mesures contenant les entrées du sujet (S), les premiers contre-sujets (CS1) aux voix de Soprano (S), Alto (A) et Tenor (T). La position de chaque occurrence est donnée par le numéro de mesure dans laquelle elle apparaît (3 est le premier temps de la mesure 3). De plus, le début est décalé par rapport à sa position logique (une croche pour S, une double croche pour CS1, sauf pour quelques occurrences où il y a un mot clé "start" supplémentaire). Cette fugue contient également des cadences parfaites et imparfaites (PAC : Perfect Authentic Cadence et IAC : Imperfect Authentic Cadence) à la tonique (i, I) et à la dominante (v). Les majuscules désignent les tonalités majeures et les minuscules les tonalités mineures. (Bas) Visualisation de ce fichier des analyses de référence, et résultats du système proposé pour la même fugue. Les algorithmes trouvent toutes les entrées du sujet (et, en chiffres romains, les degrés du sujet par rapport à la tonalité globale), les occurrences des "têtes" de sujet (boîtes non étiquetées à l'intérieur des épisodes), toutes les entrées du premier et du deuxième contre-sujet sauf une et les marches harmoniques (s), 3 cadences sur 4 et la pédale finale. Les cadences sont désignées par leur fondamentale. L'algorithme ici a une précision parfaite.

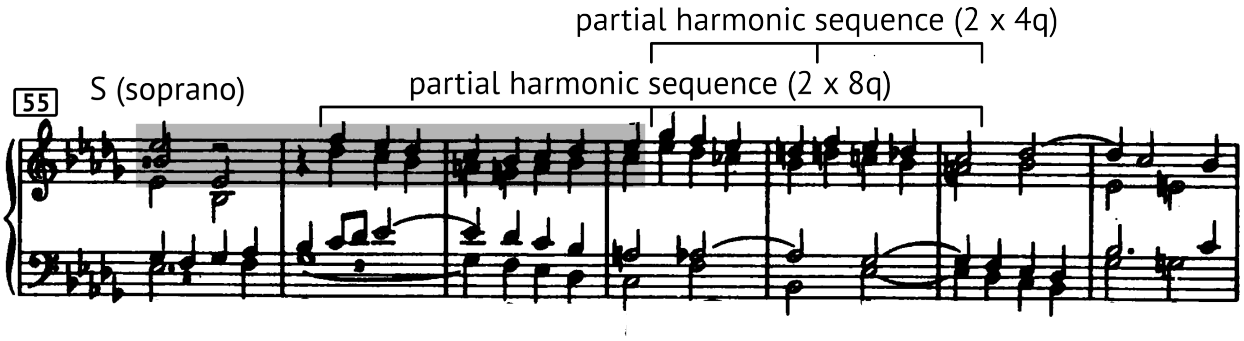


FIGURE 8.6 – Fugue n° 22 en Si bémol mineur. Le sujet est présenté à la soprano aux mesures 55, 56, et 57. La ligne de tierces entre les voix S et A donne le motif de l'épisode suivant. Une marche harmonique partielle est détectée commençant à la mesure 56 pour 2 fois 2 mesures (8 noires) – cette suite est approchée, car la fin du motif est variée. Une autre marche harmonique partielle avec 2 fois 1 mesure est détectée plus tard.

pédales se trouvent à la toute fin des fugues, comme dans la fugue n° 2 (voir la figure FIGURE 8.7). On comptabilise comme "pédale de basse" toute note de la basse qui dure strictement plus que 4 temps (pour des métriques binaires) ou six temps (pour des métriques ternaires).

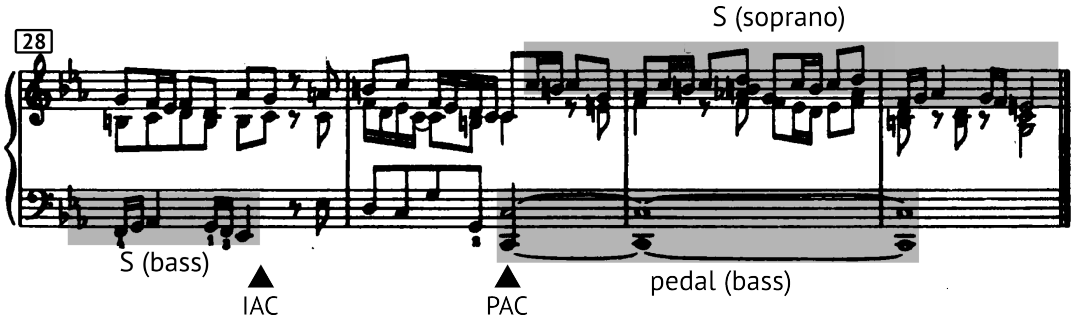


FIGURE 8.7 – Fin de la fugue n° 2 en Do mineur de J.-S. Bach, BWV 847). La cadence parfaite conclusive (PAC, milieu de la mesure 29) est suivie par une pédale de basse et une dernière exposition du sujet par la voix soprano.

Chaque fois qu'il y a un mouvement de basse d'une quarte parfaite supérieure ou d'une quinte parfaite inférieure, on essaie d'identifier une cadence V-I en position fondamentale si les conditions suivantes sont vérifiées : *Sur le temps*, les notes de l'accord doivent faire partie d'un accord de quarte majeure, parfait ou suspendu (sus4) sur la tonique de la basse ; la note supérieure doit être soit la tonique (cadence parfaite, PAC) ou une tierce majeure ou mineure (cadence imparfaite – avec renversement –, rIAC). Notons que l'accord d'arrivée n'est souvent pas un accord parfait complet, et parfois il peut même être limité à une seule note (la tonique), comme dans la PAC de la figure 8.7. *Avant le temps*, à un moment pendant la note de basse, on doit trouver l'accord de dominante (notes inclues dans une septième de dominante, avec au moins le ton principal). À cause des notes de passage, cet instant ne correspond pas en général à la toute dernière note avant le temps fort.

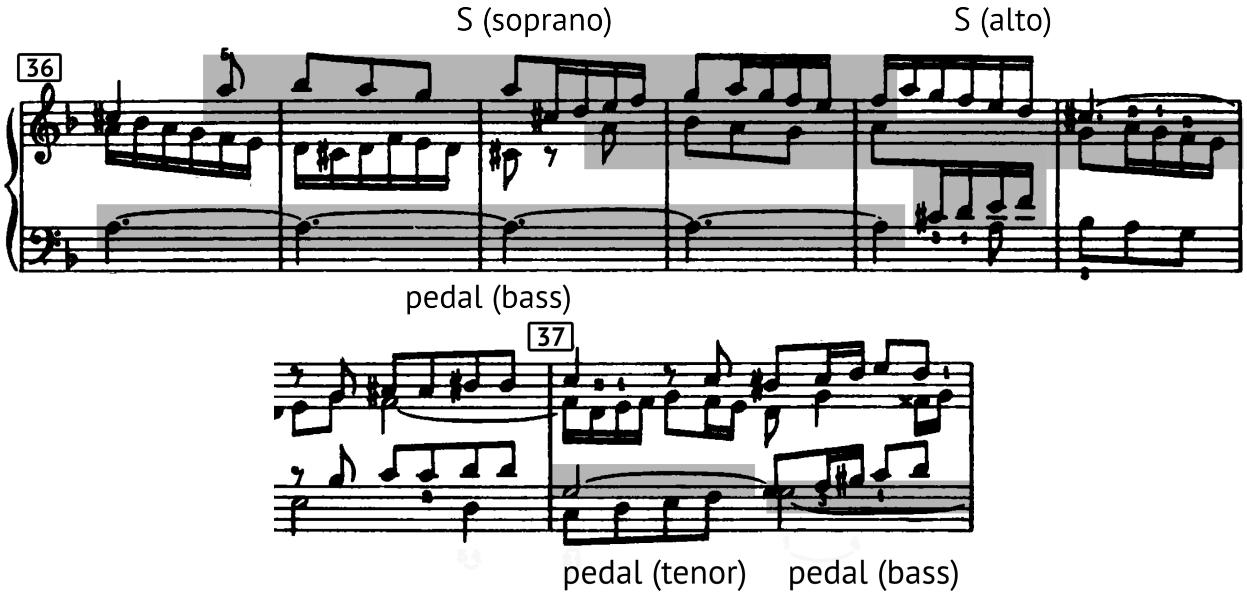


FIGURE 8.8 – Cas inhabituels de pédales. (a) Au milieu de la fugue n° 11, en Fa majeur, il y a une pédale (correctement détectée) à la dominante de la tonalité relative. (b) Cette pédale dans la figue n° 18 en Sol dièse mineur, commence au ténor et continue sur la basse.

Structure globale

On utilise un modèle de Markov caché (*hidden Markov model –HMM*) pour combiner les différents éléments de l'analyse. Cela consiste en une suite d'*états* avec des *probabilités de transitions* et des *probabilités d'émission*. À chaque étape du calcul, le modèle peut changer son état en suivant une transition. Il émet des symboles, qui sont les seules sorties du système. La suite des états correspondant est cachée. Pour une suite de symboles observée, l'algorithme de Viterbi trouve la suite d'états la plus probable par programmation dynamique [95].

Dans l'analyse de la fugue, l'objectif est de trouver la suite d'états qui structure la fugue (exposition, codetta, suite de l'exposition, épisodes, voir la figure 8.10). Les symboles observés sont les éléments de base de l'analyse (entrée des motifs thématiques, marches harmoniques) pour chaque temps de la partition. Un exemple de sortie du modèle se trouve au bas de la figure 8.5.

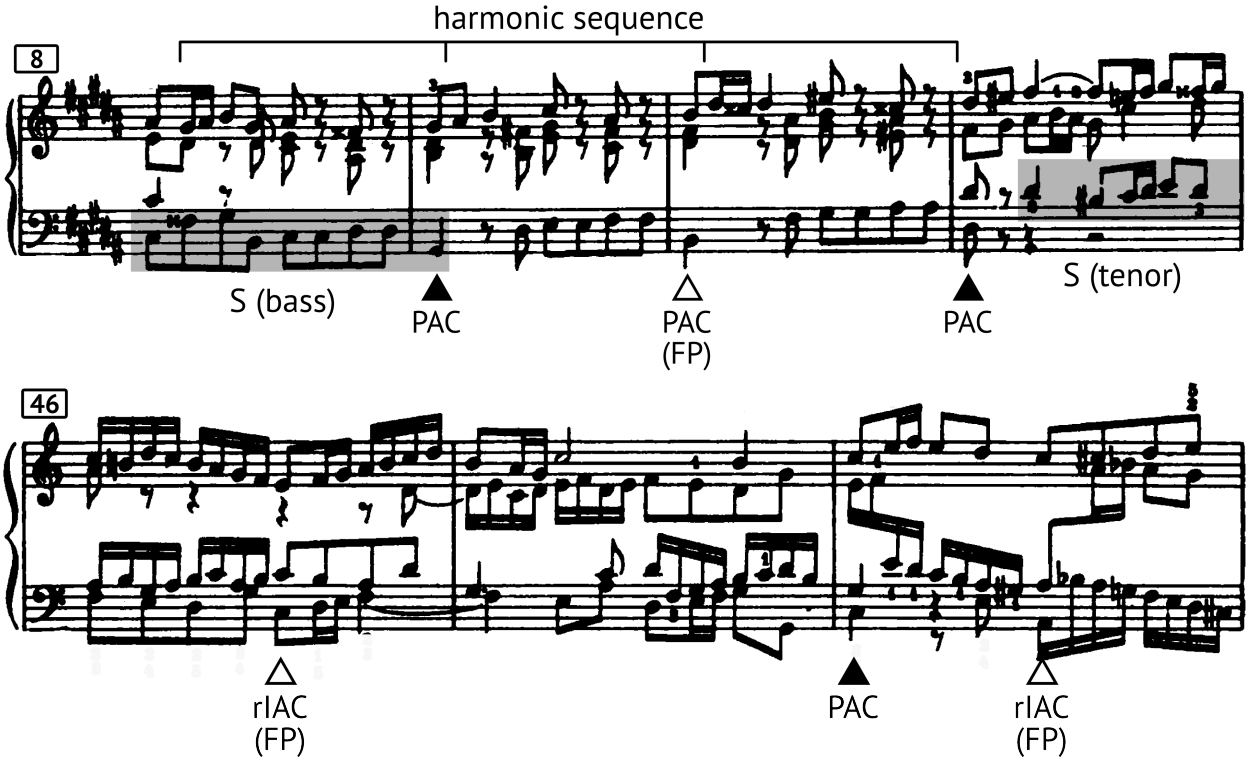


FIGURE 8.9 – Exemples de cadences trouvées. (a) Fugue n° 18, en Sol Dièse mineur, mesures 8 à 11. Il y a un intervalle de quinte parfaite à la fin du sujet, menant à des cadences erronées quand le sujet est donné par la basse. L'algorithme détecte ici trois cadences, au premier temps de chaque mesure. La première (premier temps mesure 9), peut être vue comme une cadence parfaite finissant l'exposition des motifs thématiques. Celle du milieu (premier temps de la mesure 10) est un faux positif. La dernière (premier temps de la mesure 11) peut être vue comme une cadence parfaite terminant la marche harmonique. Elle est renforcée par le changement de texture à la mesure 11, avec une nouvelle entrée du sujet au ténor sur le deuxième temps. (b) Fugue n° 20 en La mineur, mesures 46 à 48. L'algorithme proposé trouve trois cadences consécutives, mais seule celle sur le premier temps de la mesure 48 est une cadence parfaite, les notes sur le troisième temps des mesures 46 et 48 faisant partie d'une phrase plus longue.

8.3 Vers une analyse automatique des formes sonates

Le contenu de cette partie a été publié en 2014 dans [DGGLL2014], en collaboration avec Mathieu Giraud, Richard Groult, ainsi que Laurent David, étudiant en dernière année à l'ENSEA de Lille et Corentin Louboutin, étudiant en L3 à l'ENS Lyon.

8.3.1 La forme sonate

La forme sonate est une forme musicale classique développée à partir de la fin du XVIII^{e} siècle jusqu'à la fin du XIX^{e} siècle. Provenant initialement d'une forme binaire qui a été étendue, cette forme comporte trois parties jouant des rôles musicaux complémentaires (figure 8.11):

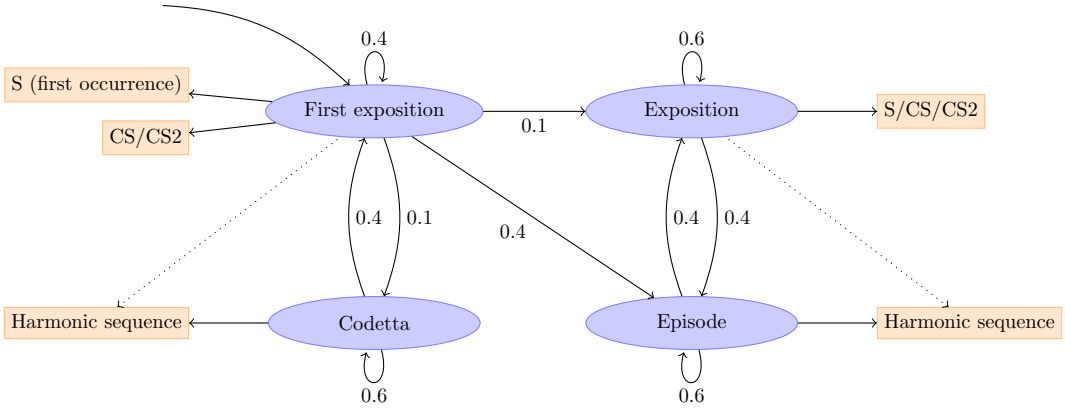


FIGURE 8.10 – Schéma du Modèle de Markov caché utilisé pour les fugues. Les probabilités de transitions ont été sélectionnées empiriquement pour favoriser une certaine stabilité des états, les sections durant en général au moins une mesure. Les états de première exposition et de codetta sont utilisés seulement pendant l'exposition. Les émissions en flèches pointillées sont rares, avec de faibles probabilités (comme les marches harmoniques pendant des parties d'exposition, par exemple).

— *L'exposition* a un rôle à la fois mélodique et harmonique, permettant à la fois d'exposer les thèmes (deux en général, principal et secondaire) qui seront utilisés dans les autres parties et d'affirmer la tonalité principale de la pièce. En général, le *thème principal* (P) est exposé à la tonique (I) puis, après une transition et une césure médiane (*medial caesura*, MC), le *thème secondaire* (S) est exposé dans une autre tonalité, généralement au ton de la dominante (V) pour les formes sonates en mode majeur.

— *Le développement* réutilise les thèmes présentés dans l'exposition tout en créant une tension tonale. Présentant de nombreuses modulations, c'est une zone tonalement instable.

— Enfin, *la réexposition* (*recapitulation* en anglais) présente à nouveau les thèmes principaux. Cette fois-ci, le deuxième thème est joué à la tonique, ce qui permet une affirmation de la tonalité principale de la pièce.

La figure 8.12 montre un exemple d'une forme sonate très simple, dans une sonatine pour piano de Kuhlau.

Plusieurs marqueurs se trouvent entre les différentes parties, en particulier des cadences (fins de phrases musicales). Le thème principal présenté dans l'exposition est souvent suivi d'une transition se terminant par une *demi-cadence* (*half cadence*, HC). Le thème secondaire se termine généralement par une *cadence parfaite* (*Perfect Authentic Cadence*, PAC), qui conclut musicalement l'exposition. Enfin, l'exposition se termine par une *conclusion* (qui ne présente généralement pas de nouvel élément thématique). Les cadences et la conclusion, parfois transformées, se retrouvent dans la réexposition.

Il y a beaucoup de variations possibles sur cette structure, selon l'époque et les compositeurs. Haydn utilise par exemple beaucoup de formes sonates mono-thématiques, comme dans l'op. 33 n° 2. En fait, la forme sonate ne se caractérise pas par la succession des sections, mais plus par un *équilibre à grand échelle* : les tensions tonales (autre tonalité) et rhétoriques (thème S, texture) créées lors de l'exposition et étendues dans le développement finissent par se résoudre dans la réexposition.

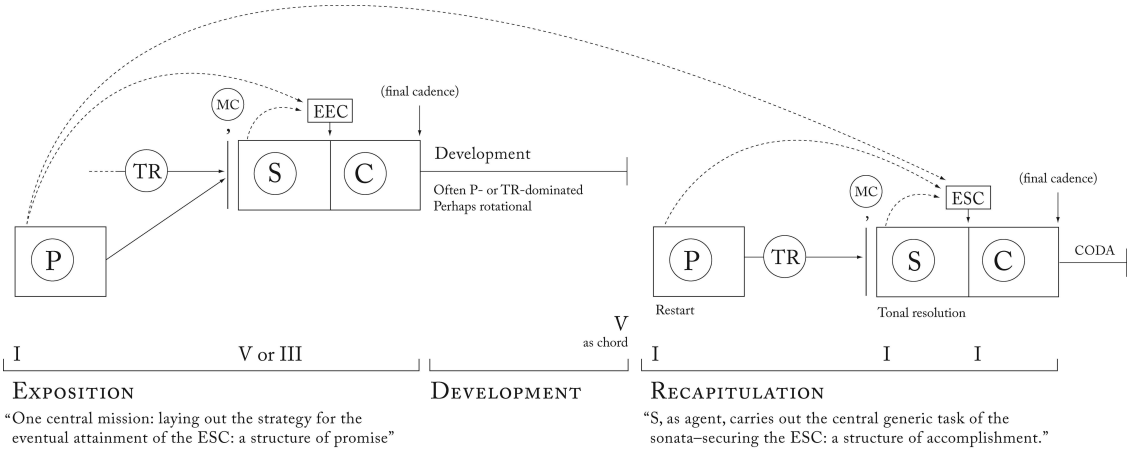


FIGURE 8.11 – Structure d'ensemble d'une forme sonate, schéma et notations tirés du livre de Hepokoski et Darcy [40, page 17]. La forme sonate est construite sur un thème (ou une zone thématique) principal (P), un thème secondaire (S) et une conclusion (C). Entre les deux thèmes, la transition (TR) se termine par la césure médiane (*medial caesura*) (MC). À la fin du thème secondaire se trouve une cadence parfaite, dénommée EEC (*Essential Expositional Closure*) dans l'exposition et ESC (*Essential Structural Closure*) dans la réexposition (*recapitulation*).

8.3.2 Analyser informatiquement la forme sonate ?

La communauté MIR (Music Information Retrieval) s'intéresse à l'analyse musicale dans son ensemble, à partir de données audio ou symboliques. Bien que l'analyse structurelle soit au cœur de ces thématiques, à ce jour, peu de travaux en MIR se concentrent sur les formes sonates. Même si les formes sonates servent de support à l'extraction de motifs (voir par exemple [73]), leur analyse à grande échelle est peu abordée. Dans un article récent, Jiang et Müller effectuent des comparaisons de signaux audio pour retrouver le couple exposition/réexposition sur les premiers mouvements dans 28 sonates pour piano de Beethoven [47]. Ils proposent aussi de suivre les changements harmoniques à l'intérieur des différentes parties.

La donnée symbolique de la partition devrait permettre d'être bien plus précis dans l'analyse. Nous pensons que l'analyse des formes sonates est un défi intéressant pour la recherche en informatique musicale symbolique, demandant de combiner des aspects locaux (motif et thèmes, cadences...) avec des aspects globaux (plan tonal, structure d'ensemble). Le premier défi est la *détection de la structure générale*: couple exposition/réexposition, avec des marqueurs précis lorsqu'ils existent, formant la structure tonale à grande échelle. Le second est de retrouver la *structure tonale et motivique du développement*. Le développement montrant souvent une grande instabilité des tonalités, il serait intéressant de pouvoir décrire de manière cohérente ce parcours tonal. Il serait également intéressant de trouver quels éléments thématiques de l'exposition sont utilisés et sous quelle forme ils sont repris, la difficulté étant qu'il peut s'agir de très courts motifs mélodiques ou rythmiques ou qu'ils peuvent présenter des variations conséquentes.

Toute approche automatisée doit être conçue en gardant à l'esprit les limites liées à la subjectivité de toute analyse – cependant, certains points simples d'analyse peuvent faire consensus et permettre d'évaluer les algorithmes.

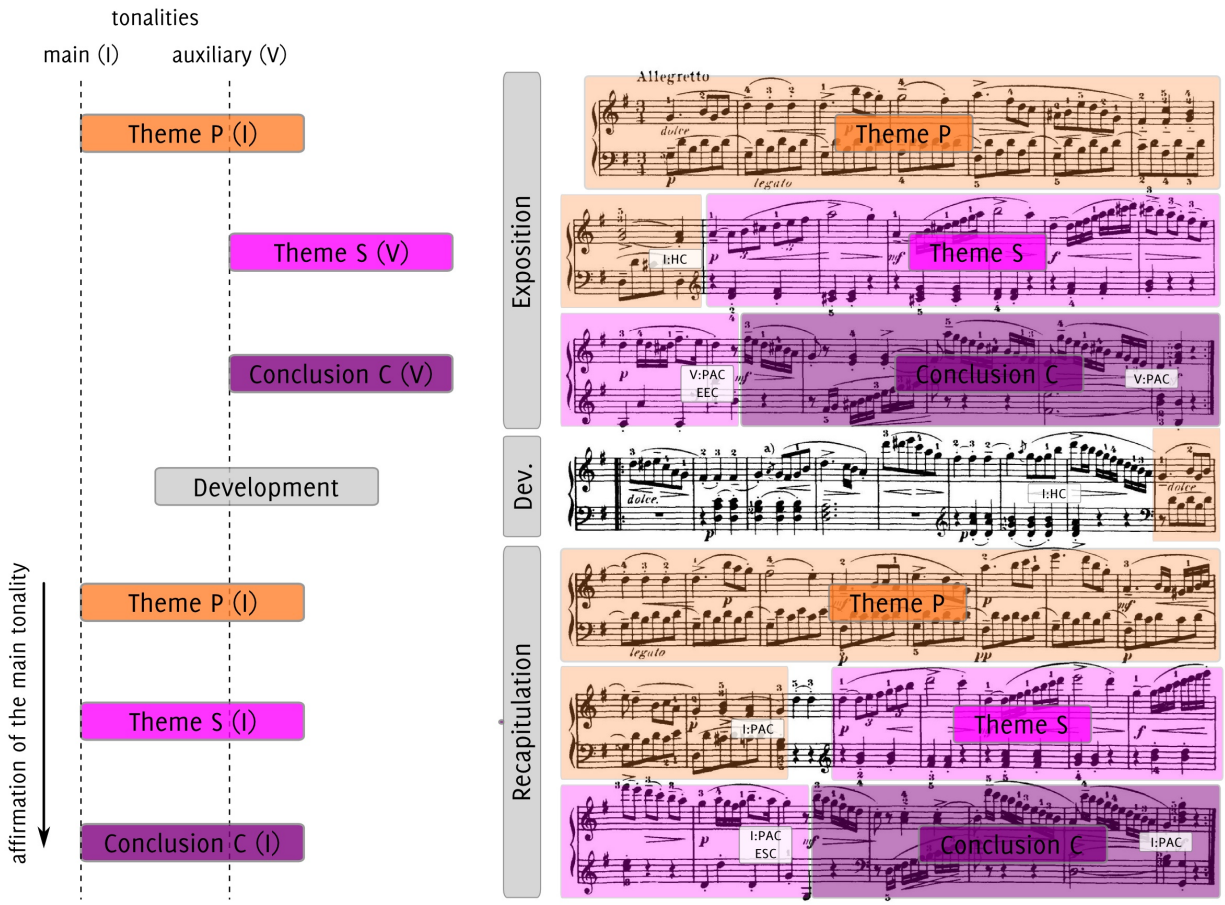


FIGURE 8.12 – Analyse structurelle de l'Allegretto de la sonatine Op. 55, n° 2 de Kuhlau, pour piano, en utilisant les conventions de [40]. Les cadences (fin de phrases) sont indiquées par les termes HC (*Half Cadence*, demi-cadence) et PAC (*Perfect Authentic Cadence*, cadence parfaite), précédés de la tonalité de cette cadence (ici I et V). Les termes EEC et ESC désignent les fins de sections (voir figure 8.11). Cette sonatine est très simple, avec un développement très court (Dev.), et pratiquement aucune transition entre les zones thématiques P et S. Le groupe thématique P dure 8 mesures lors de l'exposition et 11 lors de la réexposition, ce qui renforce la stabilisation de la tonique dans la réexposition. Le thème S et la conclusion C sont exactement transposés entre l'exposition et la réexposition.

8.3.3 Détection du couple Exposition/Réexposition

Nous décrivons ici quelques expériences sur l'analyse informatique de formes sonates, en partant de partitions sous forme de données symboliques. Nous partons de fichiers Humdrum [44], qui encodent les partitions plus précisément que des fichiers MIDI (avec notamment les informations d'enharmoniques), mais nous nous limitons dans cette étude aux informations de hauteur et de durée.

Pour ces expériences, nous nous sommes limités à des quatuors à cordes, ce qui facilite l'analyse étant donné que les pièces sont séparées en quatre voix (premier et deuxième violon, alto, violoncelle). Nous avons ainsi rassemblé 11 premiers mouvements de quatuors de Mozart, Haydn et Schubert (figure 8.14).

Afin d'évaluer nos algorithmes, nous avons manuellement identifié pour chacun de ces mouvements le couple exposition/réexposition, en nous appuyant sur les analyses de Hepokoski et Darcy [40] ainsi que sur notre propre analyse.

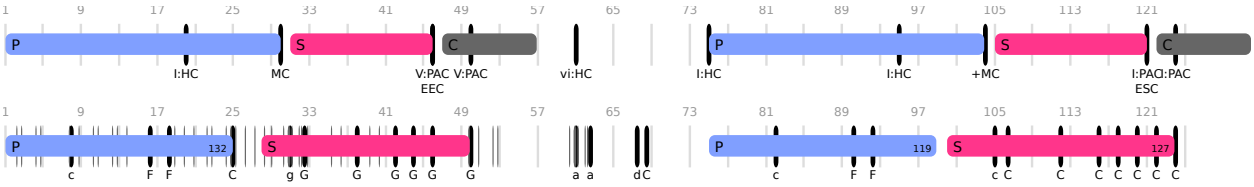


FIGURE 8.13 – Recherche du couple exposition/réexposition dans le premier mouvement du quatuor à cordes K. 157 n° 4 de W. A. Mozart comprenant 126 mesures. Le schéma du haut montre une analyse possible (faite par nos soins) avec cadences et marqueurs structurels en utilisant les notations de [40] (voir figure 8.12). Le schéma du bas représente ce qui est trouvé par le programme. Les traits fins verticaux dans l'exposition montrent des fins de phrases potentielles, et les traits gras des cadences potentielles (majuscule: majeur, minuscule: mineur). Le couple exposition/réexposition est retrouvé, mais la zone S prédite est détectée en amont de la zone réelle S, la fin de la transition entre P et S étant déjà transposée à la dominante dans l'exposition. De plus, la zone S prédite s'étend aussi sur la conclusion C.

La figure 8.13 montre une analyse obtenue sur le quatuor K. 157 n° 4 de Mozart avec les opérations suivantes :

— recherche de cadences parfaites (V vers I en position fondamentale, avec arrivée sur la tonique à la voix supérieure) [GGLL2015] ;

— recherche de fins potentielles de phrases par une simple détection rythmique (notes plus longues que leur entourage ou suivies de silences) ;

— recherche des zones P/S, par une adaptation de l'algorithme de Mongeau-Sankoff qui calcule, par programmation dynamique, les similarités entre deux mélodies [68]. Nous utilisons une fonction de score qui favorise les similarités diatoniques [GGL2012b]. Cette recherche concerne uniquement la voix supérieure (premier violon).

La zone P est recherchée du départ à une fin de phrase, sans transposition. La zone S est recherchée également après la zone P, sous la contrainte d'une transposition de la dominante vers la tonique (V vers I). Dans ce mouvement du quatuor K. 157, on voit que la stratégie met en évidence le couple exposition/réexposition.

La figure 8.14 montre les résultats sur les 11 pièces. Le début de P, marquant le début du couple exposition/réexposition, est retrouvé 9 fois à la position correcte, dont 7 fois de manière unique. Dans 8 de ces 9 cas, une deuxième zone (S) est identifiée avec une transposition de V vers I, dans la plupart des cas à moins de 10 mesures du début réel du thème S :

— la fin de la transition entre P et S peut être identique (mais transposée) entre l'exposition et la réexposition (prédiction de S avant le S réel) ;

— le thème S peut être transformé ou repris à une autre voix (prédiction de S après le S réel).

Certains thèmes ne sont pas identifiés ou détectés à des positions incorrectes (2 thèmes P et 3 thèmes S). Par exemple, dans le quatuor K. 80, on ne trouve pas le thème P car seule la fin est reprise dans la réexposition. Dans le quatuor K. 387, le thème S est joué par le *deuxième* violon, ce qui empêche sa détection correcte.

Ainsi, nos expériences sur quelques pièces montrent qu'il est possible informatiquement d'analyser des partitions de formes sonates en retrouvant certains éléments du couple exposition/réexposition (début de la réexposition, parcours tonal à grande échelle). Plusieurs perspectives pour progresser vers une analyse automatisée sont présentées dans le chapitre 10.

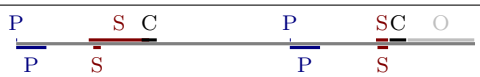
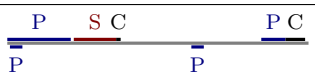
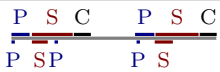
	ℓ	Analyses	P	S
Haydn op. 33 n° 2	90m 4/4		+	
Haydn op. 33 n° 3	167m 4/4		=	+8
Haydn op. 33 n° 5	305m 2/4		+	+3
Haydn op. 54 n° 3	189m 2/2		=	+13
Haydn op. 64 n° 4	99m 4/4		−	
Mozart K. 80 n° 1	68m 3/4		−	+4
Mozart K. 155 n° 2	119m 4/4		+	+3
Mozart K. 156 n° 3	184m 3/8		+	0
Mozart K. 157 n° 4	126m 4/4		+	−2
Mozart K. 387 n° 14	170m 4/4		=	−6
Schubert op. 125 n° 1	255m 2/2		+	−23

FIGURE 8.14 – Recherche du couple exposition/réexposition sur 11 premiers mouvements de quatuors à cordes en forme sonate. Pour chaque pièce, la colonne "ℓ" indique la longueur en mesures ainsi que le mètre. La colonne "Analyses" indique, en haut, quelques éléments de notre analyse manuelle pour l'exposition comme pour la réexposition (P: zones thématique principale, éventuellement suivie d'une transition, S: zone thématique secondaire, C: conclusion, O: coda), et, en bas, l'analyse effectuée informatiquement. La colonne "P" indique si le thème est trouvé exactement (+), avec quelques mesures de décalage dues à la présence d'une introduction (=) ou mal ou non trouvé (−). Lorsque le second thème est trouvé, la colonne "S" indique le décalage, en mesures, entre cette prédiction et le "vrai" second thème. Notons que le quatuor op. 33 n° 2 de Haydn est monothématique, les deux zones P et S étant construites sur le même matériau mélodique. De plus, dans les quatuors op. 54 et 64 de Haydn, il n'y a pas de réexposition pour S.

Chapitre 9

Analyse à partir de fichiers polyphoniques

Les analyses présentées dans les sections précédentes ont été réalisées à partir de fichiers dont les voix étaient séparées, c'est-à-dire sur des lignes monophoniques. Il existe des algorithmes (notamment géométriques) permettant de faire de la recherche de motifs sur des fichiers polyphoniques [67], cependant le travail direct à partir de fichiers polyphoniques est très complexe, en particulier pour de la recherche approchée. C'est pourquoi nous avons décidé de travailler sur les algorithmes de séparation en voix (permettant d'obtenir des lignes monophoniques) et en streams (regroupant des notes par groupes cohérents, possiblement polyphoniques, par exemple en suites d'accords). Ainsi nous avons co-encadré (avec M. Giraud et R. Groult) la thèse de Nicolas Guiomard-Kagan, débutée en 2013 et sur le point d'être soutenue. Nicolas a d'abord fait un état de l'art d'une part des algorithmes existants pour traiter les fichiers symboliques polyphoniques, d'autre part des méthodes de séparation de voix. Il a ensuite amélioré les algorithmes de séparation de voix à partir de fichiers polyphoniques symboliques, et proposé des métriques pertinentes pour comparer les résultats des différents algorithmes. Ses travaux ont déjà fait l'objet de deux publications en conférence internationale en collaboration avec Mathieu Giraud, Richard Groult et moi-même [GGGL2015, GGGL2016]. Ces travaux ne sont pas détaillés ici.

Dans ce chapitre, nous abordons deux problèmes liés à une certaine compréhension de la complexité de la musique polyphonique, et qui peuvent apporter des critères complémentaires pour l'étude de la structure présentée dans le chapitre précédent : le rythme et la texture.

9.1 Extraction du rythme dans la musique polyphonique

Ce travail a fait l'objet de ma première collaboration avec M. Giraud et R. Groult, par l'encadrement de stage de master de C. Séguin, et a donné lieu à un article présenté en conférence internationale [AGGGLS2011].

Dans cette section nous nous concentrons sur le composant rythmique de la similarité musicale symbolique, proposant plusieurs moyens d'extraire une signature monophonique à partir d'une partition polyphonique. La question en filigrane est de savoir si le rythme seul peut être représentatif d'un motif, et de façon générale, si le rythme pourrait aussi caractériser une chanson ou un genre musical, et être un moyen d'identifier une chanson. Les exemples de rythmes particulièrement

caractéristiques qui viennent en général à l'esprit sont le début de la *5ème symphonie* de Beethoven, le motif de batterie de *We Will Rock You* ou le *Boléro* de Ravel. Les études sur le "Query By Tapping (QBT)", où l'utilisateur tape un rythme sur un micro par exemple [46, 37], permettent dans certaines situations d'identifier une chanson monophonique. À une plus large échelle, les musicologues ont étudié comment le rythme, comme la tonalité, peut structurer une pièce à différents niveaux [21, 90].

Pour aller plus loin qu'une simple extraction de tous les intervalles de temps entre onsets, nous avons sélectionné les notes selon leur durée (courtes ou longues) et leur intensité. Une fois le rythme extrait, nous avons utilisé la programmation dynamique pour comparer plusieurs séquences.

Nous avons cherché les motifs rythmiques spécifiques à une pièce unique en fonction de leur taille, et effectué des expériences de requêtes de similarité sur du Ragtime et des variations de Chorals de J.-S. Bach. Les résultats ont montré que les extractions de notes longues et celles prenant en compte l'intensité étaient souvent de bons choix pour l'extraction de rythme. Nos conclusions sont que, même pour une musique polyphonique, le rythme seul, extrait de manière adéquate, peut suffire à identifier une pièce ou effectuer des requêtes de similarité pertinentes.

9.1.1 Représentation de séquences rythmiques monophoniques

Partant de partitions symboliques, nous supposons que les rythmes sont déjà quantisés sur les temps, et nous n'étudions pas les paramètres de tempo et de métrique. Si nécessaire, plusieurs requêtes permettent de gérer les cas où le tempo est doublé ou divisé par deux.

On modélise le rythme comme une succession de durées entre notes, c'est-à-dire des intervalles inter-onsets, mesurés en noires ou en fractions de noires (voir la figure 9.1).

FIGURE 9.1 – La séquence de rythme monophonique (1, 0.5, 0.5, 2).

Ainsi, dans ce modèle simple, il n'y a pas de silences, puisque chaque note, sauf la dernière, est considérée jusqu'au début de la note suivante.

9.1.2 Extraction de rythme

Dans cette section, nous proposons plusieurs extractions rythmiques. La figure 9.2 présente un exemple appliquant ces extractions sur le début d'un choral de J.-S. Bach.

L'extraction la plus simple est de considérer tous les onsets d'une pièce, réduisant la polyphonie à une piste monophonique simple. Cette extraction ("*noteson* extraction") extrait les durées des intervalles inter-onsets de tous les groupes de notes consécutifs. Pour chaque note ou chaque groupe de notes jouées simultanément, la durée considéree est l'intervalle de temps entre l'onset du groupe de notes courant et l'onset suivant. Chaque groupe de notes est pris en compte et représenté dans le motif rythmique extrait. Cependant, cette extraction n'est pas vraiment représentative de la polyphonie : quand plusieurs notes de durée différente sont jouées en même temps, il peut y avoir des notes plus importantes que d'autres.

En nous inspirant de l'extraction mélodique symbolique où l'on peut extraire la hauteur la plus haute (ou la plus basse) de chaque groupe de notes [105], nous avons proposé plusieurs idées pour choisir les onsets et les durées les plus représentatives de la polyphonie.

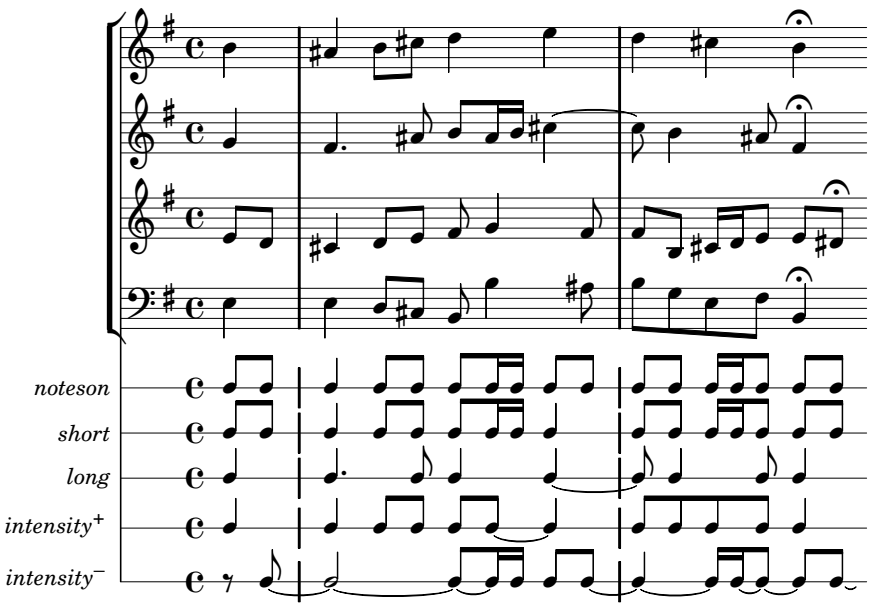


FIGURE 9.2 – Extraction de rythme sur le début du Choral de J.-S. Bach BWV 278.

Prise en compte de la durée des notes : longues, courtes À un onset donné, pour une note ou un groupe de notes jouées simultanément :

— dans l'extraction *long*, tous les évènements se produisant pendant la durée de la note la plus longue sont ignorés. Par exemple, comme il y a une noire sur le premier onset de la figure 9.2, le deuxième onset (croche au ténor) est ignoré ;

— de même, pour l'extraction *short*, tous les évènements se produisant pendant la durée de la note la plus courte sont ignorés. Cette extraction est très proche de l'extraction de base (pour tous les onsets).

Dans les deux cas, plusieurs onsets pouvant être ignorés, la durée considérée est l'intervalle de temps entre l'onset du groupe courant et le suivant non ignoré. L'extraction *long* apporte des gains significatifs dans les requêtes de similarité (voir la section 9.1.3).

Prise en compte de l'intensité des onsets La deuxième idée est de filtrer les évènements en fonction du nombre de notes se produisant, par exemple en gardant seulement les onsets impliquant au moins k notes ($intensity^+$) ou strictement moins que k notes ($intensity^-$), où le seuil k est choisi en fonction de l'intensité globale de la pièce. Les durées considérées sont les intervalles de temps entre des groupes filtrés consécutifs. La figure 9.2 montre un exemple pour $k = 3$. Cette extraction est proche de ce qui peut être fait sur le signal audio pour la détection de pics.

9.1.3 Recherche de similarités rythmiques

Protocole Notre corpus est constitué de 5900 fichiers MIDI de musique occidentale, triés par genre (dont 204 classique, 419 jazz/latin, 4924 pop/rock), où nous avons retiré la piste de batterie (channel MIDI 10) pour éviter que les extractions rythmiques ne contiennent de trop grandes suites de croches dues à une écriture répétitive. Puis, pour chaque extraction présentée ci-dessus, nous

avons traité tous les fichiers de la base. Pour chaque fichier, nous avons pris comme seuil k pour l'intensité la valeur médiane entre toutes les intensités. Pour cela, nous avons utilisé la plateforme Python `music21` [22].

Nous avons essayé d'identifier une chanson grâce à un motif de plusieurs durées successives pris dans une extraction rythmique, et cherché les occurrences de ce motif dans toutes les chansons de la base. Nous avons ensuite essayé de déterminer si ces extractions rythmiques étaient pertinentes pour détecter les similarités. Nous avons testé deux cas particuliers : le *Ragtime* (section 9.1.3) et les *variations des chorals de Bach* (section 9.1.3). Ces deux cas présentent des difficultés polyphoniques et rythmiques : le ragtime a un rythme très répétitif à la main gauche mais une main droite très libre, et les chorals de Bach présentent des différences rythmiques entre leurs différentes versions.

Comparaison de rythme monophonique Plusieurs comparaisons de rythme ont été proposées [102]. Ici, on les compare en alignant des durées. Soit $S(m, n)$ le meilleur score pour aligner localement une séquence rythmique $x_1 \dots x_m$ avec une autre $y_1 \dots y_n$. Tout comme dans la partie 8.2.2, nous calculons le score de similarité par une équation de programmation dynamique, en adaptant l'algorithme de Mongeau-Sankoff sans prendre en compte les hauteurs de notes. Comme dans les sections précédentes, l'alignement peut ensuite être retrouvé par retour en arrière dans la table de programmation dynamique.

Il peut y avoir une *correspondance* (*match*) ou une *substitution (s)* entre deux durées, une *insertion (i)* ou une *délétion (d)* d'une durée. L'opération de *consolidation (c)* est le regroupement de plusieurs durées en une durée unique, et la *fragmentation (f)* divise une durée en plusieurs durées.

Les matchs, consolidations et fragmentations respectent les temps et les temps forts de la mesure, alors que les substitutions, insertions et délétions peuvent altérer la structure rythmique et devraient être plus lourdement pénalisées. Les scores sont évalués dans la partie 9.1.3 où l'on confirme que, la plupart du temps, les meilleurs résultats sont obtenus en prenant en compte les opérations de consolidation et de fragmentation.

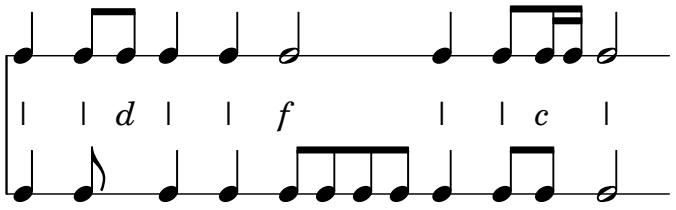


FIGURE 9.3 – Alignement entre deux séquences rythmiques.

Identification de chanson Pour chaque extraction rythmique et chaque longueur comprise entre 5 et 50, nous sélectionnons aléatoirement 200 motifs distincts apparaissant dans les fichiers de la base. Ensuite nous cherchons ces motifs dans les 5900 fichiers (figure 9.4).

On observe que dès que les longueurs grandissent, les motifs sont très spécifiques. Pour les longueurs 10, 15 et 20, le nombre de motifs (sur 200) correspondant à un fichier unique est le suivant :

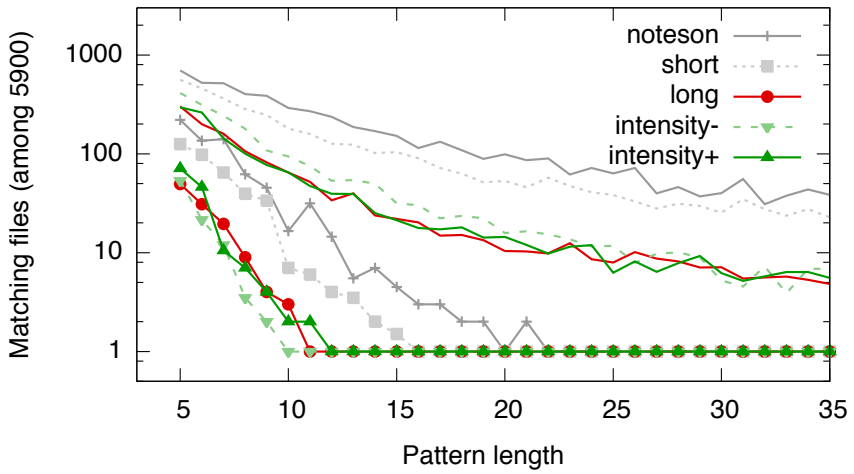


FIGURE 9.4 – Nombre de fichiers trouvés pour des motifs de longueur 5 à 35. Les courbes avec des points indiquent les valeurs médianes, les autres indiquent des valeurs moyennes.

Extraction	10 notes	15 notes	20 notes
noteson	49	85	107
short	58	100	124
long	85	150	168
$intensity^{+}$	91	135	158
$intensity^{-}$	109	137	165

On remarque que les extractions *long* et $intensity^{+/-}$ sont plus spécifiques que *noteson*. Sur 12 notes, les valeurs médianes de la figure 9.4 sont égales à 1 sauf pour les extractions *noteson* et *short*. Dans plus de 70% des requêtes, 15 notes sont suffisantes pour retrouver un fichier unique.

Les résultats pour les valeurs moyennes sont perturbés par quelques motifs correspondant à un nombre élevé de fichiers. La figure 9.5 montre quelques motifs significatifs de 10 notes. La plupart du temps, les motifs apparaissant très souvent sont des répétitions de la même note, comme le motif (a). Avec l'extraction *long*, 174 fichiers contiennent 30 noires consécutives, et 538 fichiers contiennent 30 croches consécutives. Ces nombres augmentant encore avec les extractions *noteson* (et *short*), ceci explique que l'extraction *long* soit plus spécifique.

Le nombre d'occurrences de chaque motif est déterminé essentiellement par son intérêt musical. Par exemple, dans un motif de 3 durées, (d) apparaît plus souvent que (g), qui est un rythme plutôt difficile. De la même manière, parmi les motifs contenant seulement des noires et des croches, (b) et (c) peuvent être trouvés plus souvent que (f). On remarque également que les motifs avec des durées plus longues, même répétitifs comme le motif (e), apparaissent généralement moins fréquemment que ceux contenant des durées plus courtes.

Similarités dans le Ragtime Le ragtime, l'un des précurseurs de la musique Jazz, a un tempo strict maintenu par la main gauche du pianiste et un swing typique créé par une mélodie syncopée à la main droite. Ici, nous avons considéré 17 fichiers de ragtime. Nous avons ensuite comparé certains de ces ragtimes avec un ensemble de fichiers comprenant ces 17 ragtimes et d'autres fichiers choisis

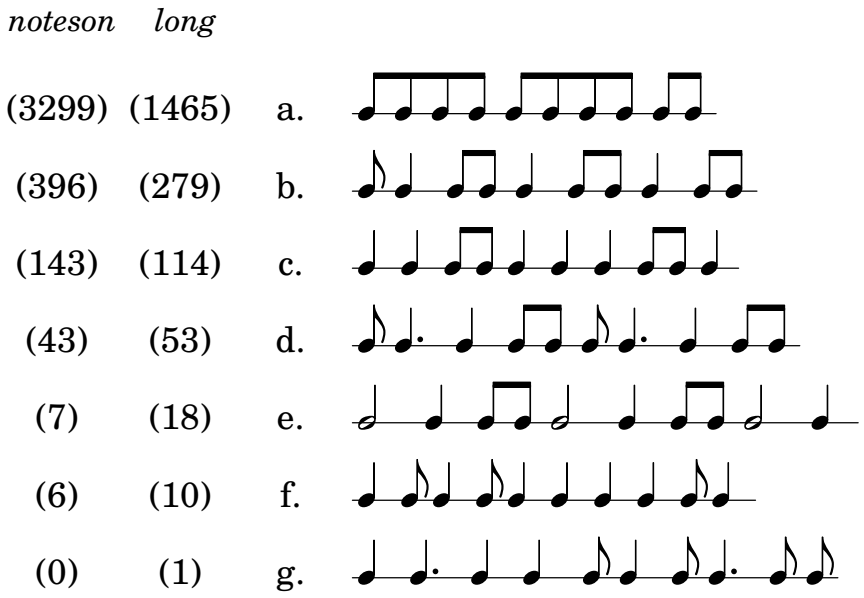


FIGURE 9.5 – Quelques motifs avec 10 durées, et le nombre de fichiers correspondant pour les extractions *noteson* et *long*.

aléatoirement dans la base. Nous avons testé plusieurs fonctions de score : toujours +1 pour une correspondance, et −10, −5, −2, −1, −1/2 ou −1/3 pour une erreur. Nous avons également testé la non pénalisation de la consolidation et de la fragmentation (c/f).

Pour chaque fonction de score, on calcule les taux de vrais et de faux positifs selon tous les seuils différents. L'extraction *long*, utilisée avec les scores +1 pour un match et −1 pour toutes les erreurs, donne ici de très bons résultats.

En moyenne sur 9 requêtes, le meilleur ensemble de scores pour chaque extraction est le suivant [1] :

Extraction	Scores			Performance moyenne
	$s/i/d$	c/f	match	
noteson	−5	0	+1	0.711
short	−1	−1	+1	0.670
long	−1	0	+1	**0.815**
$intensity^+$	−1/3	0	+1	0.622
$intensity^-$	−1	−1	+1	0.697

La plupart du temps, les séquences qui correspondent sont de longues séquences de croches, comme dans le motif (a) de la figure 9.5. Si de tels motifs sont fréquents avec une extraction *noteson*, leur présence dans les extractions longues est plus fréquente dans le ragtime que dans d'autres styles musicaux. Par exemple, le motif (a) est trouvé dans 76 % des extractions longues de Ragtimes, contre seulement 25 % dans l'ensemble de la base.

En effet, dans les partitions de ragtime, la main droite est très libre et avec beaucoup de syncopes, alors que la main gauche est mieux structurée. Ici les syncopes ne sont pas prises en compte dans l'extraction *long*, et la main gauche (souvent en croches, comme dans la figure 9.6) y est préservée.

1. Voir l'article [AGGGLS2011] pour des précisions sur le calcul de la performance

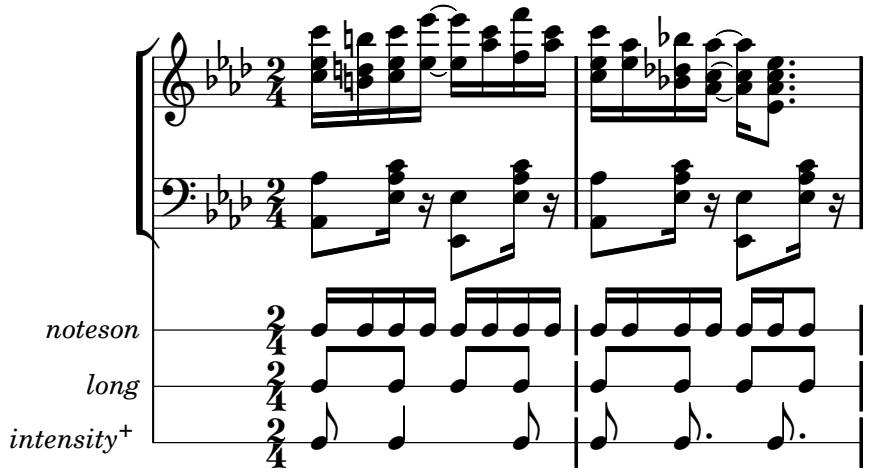


FIGURE 9.6 – *Possum Rag* (1907), de Geraldine Dobyns.

Finalement, l'extraction *intensity*$^+$ ne donne pas de bons résultats (contrairement aux Chorals de Bach, voir la partie suivante).

En fait, cette extraction préserve les syncopes de la pièce, les accents dans la mélodie impliquant souvent des accords qui passent au travers du filtre par nombre de notes (voir dans la figure 9.6).

Similarités dans les variations des Chorals de Bach Plusieurs chorals de Bach sont des variations les uns des autres, partageant une mélodie exactement identique ou très similaire.

Ces chorals présentent des variations principalement dans leur partie à 4 voix, menant à des différences dans les extractions rythmiques (voir la figure 9.7).

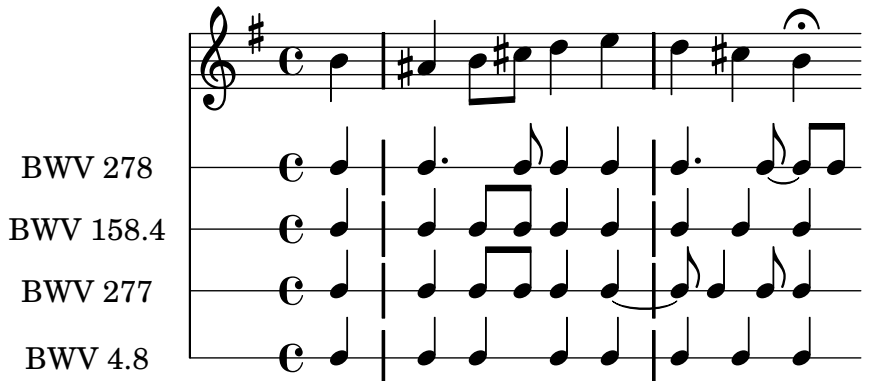


FIGURE 9.7 – Extraction de séquences rythmiques *long* de différentes variations du début du choral *Christ lag in Todesbanden*. Les différences entre les variations sont dues aux différences dans les rythmes des parties à 4 voix.

Nous considérons une collection de 404 chorals de Bach transcrits par `www.jsbchorales.net` et disponibles dans le corpus `music21` [22]. Nous avons sélectionné 5 chorals ayant plusieurs versions : *Christ lag in Todesbanden* (5 versions, dont une copie parfaite), *Wer nun den lieben Gott*

(6 versions), *Wie nach einer Wasserquelle* (6 versions), *Herzlich tut mich verlangen* (9 versions), and *O Welt, ich muss dich lassen* (9 versions).

Pour chaque choral, on utilise une version pour effectuer une requête contre tous les autres chorals, essayant de retrouver les résultats les plus semblables. En faisant une moyenne sur les 5 chorals, le meilleur ensemble de scores pour chaque extraction est le suivant [2] :

Extraction	Scores			Performance moyenne
	$s/i/d$	c/f	match	
noteson	-1	0	$+1$	0.769
short	-1	0	$+1$	0.781
long	-5	-5	$+1$	**0.871**
intensity$^+$	-1	0	$+1$	**0.880**
intensity$^-$	-5	0	$+1$	0.619

Même si l'extraction *noteson* donne déjà de bons résultats, *long* et *intensity*$^+$ amènent des améliorations notables. La plupart du temps, les meilleurs scores correspondent à des alignements entre 8 et 11 mesures, couvrant une grande partie des chorals. Nous avons ainsi réussi à aligner presque globalement un choral et ses variations. Nous avons testé s'il n'y avait pas de biais sur la longueur totale : par exemple le choral BWV 278 a une longueur d'exactement 64 temps, comme 15% des chorals, mais la distribution de score est quasiment la même dans ce choral que dans les autres.

9.1.4 Discussion

Dans toutes ces expériences, nous montrons que plusieurs méthodes sont plus spécifiques qu'une simple extraction *noteson* (ou qu'une extraction *short* similaire). L'extraction *intensity*$^-$ pourrait fournir les motifs les plus spécifiques pour utiliser comme signature (voir la figure 9.4), mais elle n'est pas appropriée pour être utilisée dans les requêtes de similarité. Les extractions *long* et *intensity*$^+$ donnent de bons résultats dans l'identification d'une chanson, mais aussi dans les requêtes de similarité au sein d'un genre ou de variations d'une musique.

Il reste à mesurer ce qui est réellement perdu en laissant de côté l'information de hauteur des notes : une perspective serait d'inclure la comparaison des extractions rythmiques avec d'autres extractions impliquant la détection de mélodie, ou l'analyse de la partie batterie supprimée dans cette étude.

Maintenant nous allons aborder une autre caractéristique susceptible d'apporter beaucoup pour l'analyse de forme en général sur des fichiers polyphoniques : la texture.

9.2 Vers une modélisation de la texture sur des données symboliques

Ce travail a été réalisé en collaboration avec M. Giraud, M. Rigaudière à l'occasion du co-encadrement de deux étudiants en Projet de Master à l'université de Lille : F. Mercier et D. Thorez. Il a fait l'objet d'une publication en conférence internationale [GLMRT2014] et la suite de ce travail est actuellement en cours avec M. Rigaudière et F. Doé de Maindreville.

2. Voir l'article [AGGGLS2011] pour des précisions sur le calcul de la performance

L'étude de la *texture* est partie intégrante du processus d'analyse musicale. En particulier, ce sont les changements de texture qui jouent un rôle important dans la connaissance des structures musicales. Si la texture est une caractéristique couramment utilisée pour analyser des données musicales audio, elle est rarement prise en compte dans les études symboliques. Nous avons proposé une formalisation de la musique instrumentale occidentale classique en couches de mélodie et d'accompagnement, et fourni un algorithme capable de détecter les couches homorythmiques dans les données polyphoniques où les voix n'étaient pas séparées. Nous présentons une évaluation de ces méthodes pour les mouvements parallèles par rapport à une analyse de référence contenant l'analyse texturale de 10 pièces instrumentales, incluant les premiers mouvements de six quatuors Op. 33 de J. Haydn.

9.2.1 Texture musicale

D'après le "Grove Music Online", la texture se réfère à *l'aspect sonore d'une structure musicale*. On différencie en général les textures *homophoniques* (parties similaires rythmiquement) des textures *polyphoniques* (différentes couches, par exemple mélodie accompagnée ou parties contrapuntiques). Certaines catégorisations plus précises ont été proposées, par exemple par Rowell [93, p. 158 – 161] qui a proposé 8 "valeurs texturales", parmi lesquels la verticalité ou l'horizontalité de l'écriture, l'entrelacement des mélodies, l'organisation en motifs, le jeu dialogique, la densité, ... Ce qui est souvent intéressant dans le discours musical est le *changement de texture* : J. Dunsby, affirme ainsi que *"on n'a pas grand chose à dire du tout sur la texture en elle-même, car tout dépend de ce qui est comparé et avec quoi"* [29].

Texture orchestrale. Le terme *texture* est utilisé pour décrire l'orchestration, c'est-à-dire la manière dont le matériel musical est organisé selon les différents instruments ou pupitres, prenant en compte les registres et les timbres. Dans son livre de 1955 sur l'*Orchestration*, W. Piston présente sept types de texture : l'unison orchestral, la mélodie accompagnée, la mélodie secondaire, l'écriture en voix, la texture contrapuntique, les accords, et les textures complexes [80].

En 1960, Q. R. Nordgren [75] pose la question : *"Est-il possible de mesurer la texture ?"*. Il propose de quantifier les relations verticales et horizontales des sons créant la texture au delà des catégories usuelles telles que homophonie/polyphonie ou léger/lourd. Il considère 8 caractéristiques, leur donnant des valeurs numériques, dont le nombre d'instruments, leur ambitus, registre et espacement, les concentrations (doublures)... Il analyse ensuite 8 symphonies de Beethoven, Mendelssohn, Schumann et Brahms avec ces critères, trouvant des différences caractéristiques entre ces compositeurs.

Texture non-orchestrale. Cependant, le terme texture se réfère aussi à la musique produite par un plus petit groupe d'instruments, même de timbre identique (comme un quatuor à cordes), ou par un unique instrument polyphonique comme le piano ou la guitare. D'un point de vue extrême, on peut considérer la texture d'un instrument monophonique : un simple suite de notes monophonique peut sonner comme une mélodie, mais peut aussi avoir le rôle de motif d'accompagnement, comme des accords arpégés ou une basse d'Alberti.

Texture en analyse musicale. L'étude de la texture fait partie d'une analyse, même si elle ne fait pas sens par elle-même. Comme établie par J. Levy, *"même si elle ne peut exister indépendamment, la texture peut rendre les relations fonctionnelles et les signes créés par d'autres variables plus*

évidents et complètement effectifs." [58]. La texture joue un rôle significatif dans la connaissance des structures musicales. J. Dunsby lui attribue deux rôles principaux : l'illusion qu'elle crée, et l'attente qu'elle fait naître auprès des auditeurs envers les textures familières [29]. J. Levy montre par de nombreux exemples que la texture peut être un signe dans la musique classique et du début du romantique, décrivant le rôle des motifs d'accompagnement, des solos et de l'unisson pour attirer l'attention de l'auditeur avant d'important changements structurels [58].

Texture et Recherche d'Informations Musicales (MIR) En général la texture n'est pas analysée en profondeur et formalisée comme d'autres paramètres (en particulier la mélodie et l'harmonie). Dans le domaine MIR, elle est souvent utilisée en analyse audio, mais réduite à une description timbrale. Toute méthode traitant de signaux audio est d'une certaine manière liée au timbre et à la texture [53, 24]. Il existe par exemple plusieurs études de la segmentation utilisant la texture audio [24, 104]. Plus généralement, le terme "texture sonore" peut être utilisé pour décrire ou synthétiser des signaux audio non-instrumentaux, comme les sons ambiants [94, 99].

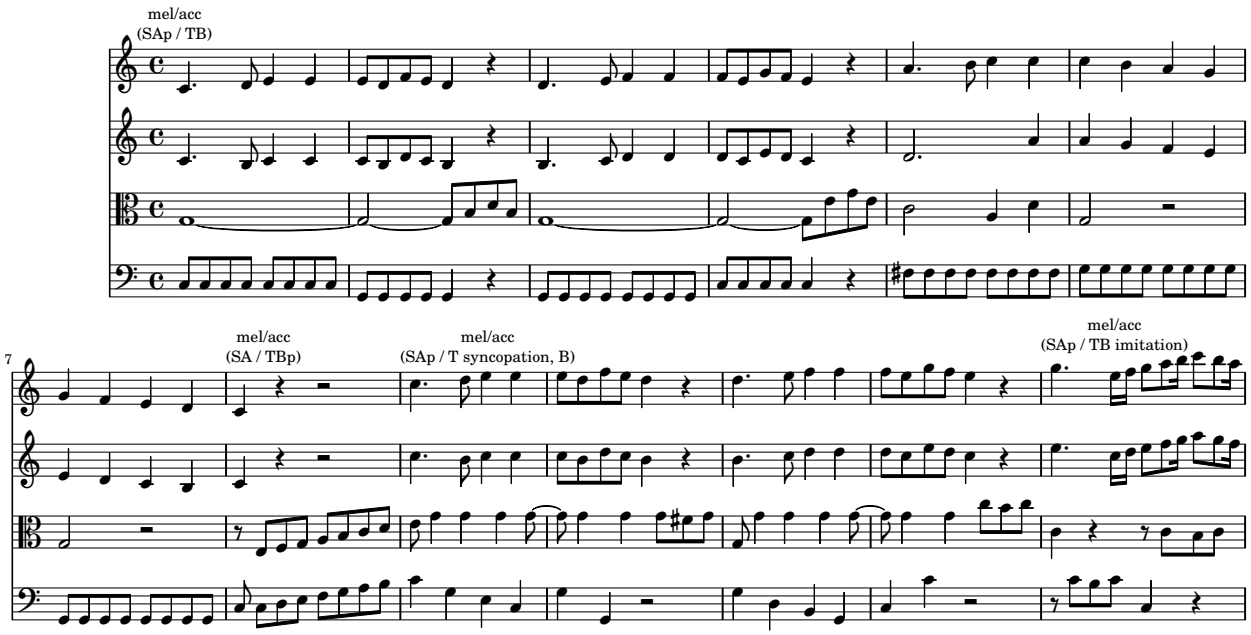
Parmi les études analysant des partitions représentées par des données symboliques, peu prennent en compte la texture. En 1989, D. Huron [43] explique que les trois significations courantes du terme texture sont le volume, la diversité des éléments utilisés et la description de la "surface", les deux premiers étant plus facilement formalisables. Utilisant un espace bidimensionnel basé sur la synchronisation des onsets et le mouvement de hauteurs, il a identifié 4 grandes catégories de textures : monophonie, homophonie, polyphonie et hétérophonie. Il a également trouvé que des genres musicaux différents occupaient différentes régions de l'espace défini.

Certaines caractéristiques de la bibliothèque jSymbolic, utilisée pour la classification de fichiers MIDI, concernent la texture musicale [65, 64]. *"[Elles] tiennent compte spécifiquement du nombre de voix indépendantes dans une pièce et comment ces voix sont liées entre elles ."* [64, p. 209]. Ces caractéristiques sont calculées sur des fichiers MIDI où les voix sont séparées, et incluent des paramètres statistiques sur l'organisation des chorals ou de la musique orchestrale : nombre de notes maximum, moyen, variabilité de paramètres sur les voix individuelles (nombre de notes, durée, dynamique, sauts mélodiques, ambitus), voix la plus grave, la plus haute, chevauchement, mouvement parallèle, distance de hauteur entre les voix.

En 2008, Tenkanen et Gualda [101] détectent les frontières articulant une pièce musicale selon 6 critères incluant des ensembles de classes de hauteur et des rations de densité sur les onsets. D. Rafailidis et al. segmentent une partition en plusieurs flux ("streams") basés sur des règles de proximité de hauteur et de temps. [15, 84].

Ainsi, il n'y a pas énormément d'études sur la modélisation ou l'analyse automatique de la texture. Même si la description de la texture musicale peut être faite à un niveau local de la partition, elle nécessite une compréhension plus haut-niveau de la musique. Nous proposons ici de ne pas prendre en compte les paramètres d'orchestration, mais de nous concentrer sur des caractéristiques données par la configuration locale des notes, prenant en compte la façon dont elles pourraient être divisées en différentes couches. De même, nous ne regardons pas l'harmonie ou les motifs, les phrases ou la répétition de motifs à large échelle.

La section suivante présente un modélisation formelle de la texture et une analyse de référence des premiers mouvements de dix quatuors à cordes. Nous proposons ensuite un algorithme pour découvrir certains éléments de texture dans des partitions polyphoniques où les voix ne sont pas séparées, et finalement nous présentons une évaluation de cet algorithme et une discussion sur les résultats.




```
 1,  75 : mel/acc (SAp / TB)
 8,  82 : mel/acc (SA / TBp)
 9,  83 : mel/acc (SAp / T syncopation, B)
13,  87 : mel/acc (SAp / TB imitation)
19,  93 : mel/acc (SAo / TBh)
20,  94 : mel/acc (SAp / TBhr)
21,  95 : imitation/acc (SA / TB)
25,  99 : imitation/acc (SA / TB)
    102 : mel/acc (S / ATB)
29, 103 : mel/acc (S / ATBhr)
30      : mel/acc (S / ATBh)
    104 : mel/acc (SAh/ TBh)
31, 105 : mel/acc (S / ATBh sparse chords)
33, 107 : acc/mel/acc (S / ATp / B)
  ...
```

FIGURE 9.8 – Début du quatuor K. 157 n° 4 de W. A. Mozart, avec une analyse de référence décrivant les textures. On étiquette S / A / T / B (soprano / alto / tenor / basse) les quatre instruments (violon I / violon II / alto / violoncelle). Les huit premières mesures ont une couche mélodique "SAp" construite selon un mouvement parallèle (en tierces), cependant le mouvement parallèle contient quelques exceptions (unisson sur la note "do", temps fort sur les mesures 1 et 8 et petite interruption au début de la mesure 5).

9.2.2 Formalisation de la texture

Modélisation en couches Nous choisissons de modéliser la texture en regroupant les notes en plusieurs "couches" (ou "streams") selon la façon dont elles sont perçues.

La ségrégation de flux d'écoute ("Auditory stream segregation") fut introduite par Bregman, qui a étudié plusieurs paramètres influençant cette ségrégation [12]. En se concentrant sur les

informations contenues dans la partition, les notes peuvent être regroupées en de telles couches selon des règles de perception [27, 84]. Le nombre de couches n'est pas directement le nombre de voix (monophoniques) jouées par les instruments, il peut y avoir jusqu'à seulement une couche perçue, lorsque plusieurs voix se fondent en homorythmie. Au contraire, certains motifs conçus sur une voix unique peuvent être perçus comme plusieurs couches, comme la basse d'Alberti.

Précisément, nous modélisons la texture en couches selon deux vues complémentaires. Tout d'abord, nous considérons deux rôles principaux pour les couches, selon leur perception par l'auditeur : les couches *mélodiques* (mel) (dominées par un mouvement de hauteurs contigües), les couches d'*accompagnement* (acc) (dominées par l'harmonie et/ou le rythme). Ensuite, nous décrivons comment chaque couche est composée :

— Une couche mélodique peut être soit une voix monophonique (solo), soit deux voix ou plus en homorythmie (h), ou avec une relation plus proche, comme (du plus générique au plus similaire) un mouvement parallèle (p), un doublement à l'octave (o) ou à l'unisson (u). Les relations h/p/o/u n'ont pas besoin d'être exactes : par exemple, un mouvement parallèle peut être partiellement en tierces, partiellement en sixtes, et inclure quelques notes étrangères (voir la figure 9.8).

— Une couche d'accompagnement peut aussi être décrite par des relations h/p/o/u, mais elle mérite souvent que l'on regarde sa composante rythmique : par exemple, une telle couche peut contenir des accords tenus, épars ou répétés, une basse d'Alberti, des notes de pédale, ou des syncopes.

Les catégories de texture usuelles peuvent alors être définies par :

— **mel/acc** – une mélodie accompagnée ;

— **mel/mel** – deux mélodies indépendantes (contrepoint, imitation...) ;

— **mel** – une mélodie (soit solo, soit plusieurs voix avec une relation h/p/o/u), pas d'accompagnement ;

— **acc** – seulement un accompagnement, quand aucune mélodie ne se dégage de ce qu'on entend (comme dans certaines transitions par exemple)

Ce formalisme permet de décrire davantage de couches, comme acc/acc, mel/mel/mel/mel, ou mel/acc/acc.

Limites. Cette modélisation de la texture est souvent ambigüe, et a des limites. On peut s'interroger sur la distinction entre mélodie et accompagnement. Certaines mélodies peuvent contenir des notes répétées, des motifs en arpèges et impliquer fortement une certaine harmonie. Il est également trop simpliste de limiter le rôle de l'accompagnement à de l'harmonie et du rythme. De plus, certains gestes texturaux ne sont pas modélisés, comme les gammes ascendantes ou descendantes. Finalement, ce que Piston appelle des "textures complexes" (et qui est sans doute le plus intéressant), à savoir l'entrelacement des différentes couches [80, p. 405], ne peut pas toujours être modélisé de cette façon. Néanmoins, la formalisation ci-dessus est fondée pour la plupart des musiques de la période classique et romantique, et correspond à un mode d'écriture mélodie/accompagnement.

Une analyse de référence pour la texture Nous avons manuellement analysé la texture de 10 premiers mouvements de quatuors à cordes les six quatuors de l'Op. 33 de Haydn, trois quatuors de jeunesse de Mozart (K. 80 n. 1, K. 155 n. 2 and K. 157 n. 4), et le quatuor Op. 125 n. 1 de

	tonalité	longueur	mel/acc	mel/mel	acc/mel/acc	acc/mel	mel	acc	autres	h	p	o	u
Haydn Op. 33 n. 1	Si m	91m	38	0	8	1	0	0	5	19	21	1	0
Haydn Op. 33 n. 2	Mi♭ M	95m	37	0	2	4	0	0	7	34	13	0	0
Haydn Op. 33 n. 3	Do M	172m	68	0	0	0	3	13	6	29	50	1	0
Haydn Op. 33 n. 4	Si♭ M	90m	25	0	1	0	0	0	6	16	6	0	0
Haydn Op. 33 n. 5	Sol M	305m	68	0	3	4	7	0	5	56	45	6	0
Haydn Op. 33 n. 6	Ré M	168m	58	0	1	3	15	0	29	43	42	0	2
Mozart K. 80 n. 1	Sol M	67m	36	4	6	0	2	0	3	5	33	3	0
Mozart K. 155 n. 2	Ré M	119m	51	0	0	0	1	0	0	21	32	4	1
Mozart K. 157 n. 4	Do M	126m	29	0	3	6	2	0	7	18	22	2	0
Schubert Op. 125 n. 1	Mi♭ M	255m	102	0	0	0	20	2	0	54	8	46	2
		1488m	512	4	24	18	50	15	68	295	272	63	5

TABLE 9.1 – Nombre de segments dans l'analyse de référence des dix quatuors à cordes (premiers mouvements) et nombre d'étiquettes h/p/o/u décrivant ces couches.

Schubert. Ces pièces couvrent les caractéristiques texturales que nous souhaitons expliciter. Nous avons segmenté chaque pièce en segments non chevauchants basés uniquement sur l'information texturale, en utilisant le formalisme décrit ci-dessus.

Il est difficile de s'accorder sur la signification de segments courts et sur leurs frontières. Ici nous avons choisi de reporter la texture avec une résolution d'*une mesure* : on considère seulement les segments durant au moins une mesure (ou remplissant la majeure partie d'une mesure) et on arrondit les frontières de ces segments aux barres de mesure.

Nous avons identifié 691 segments dans les 10 pièces, la table 9.1 détaille la répartition de ces segments. Le fichier contenant l'analyse de référence est disponible sur `www.algomus.fr/truth`, et la figure 9.8 montre l'analyse du début du quatuor K. 157 n. 4 de Mozart.

On précise pour chaque segment les voix impliquées et les relations h/p/o/u. Par exemple, si on considère la catégorie la plus représentée : "mel/acc", il y a 254 segments étiquetés soit "S / ATB" ou "S / ATBh" (couche mélodique au premier violon) et 81 segments étiquetés "SAp / TB" ou "SAp / TBh" (couche mélodique aux deux violons, en mouvement parallèle). Notons que les relations h/p/o/u ont été évaluées de façon subjective. Les segments peuvent contenir de courtes interruptions qui n'altèrent pas la perception générale de la relation h/p/o/u.

9.2.3 Trouver les couches synchronisées

Notre objectif est donc de proposer une analyse de la texture à partir d'une partition polyphonique où les voix ne sont pas séparées.

Une première idée est de commencer par segmenter la partition en couches avec les critères de perception ci-dessus, puis d'essayer de qualifier certaines de ces couches. On peut par exemple utiliser l'algorithme de [84] pour segmenter les pièces musicales en couches (appelées "streams"). Cet algorithme repose sur une matrice de distance, indiquant pour chaque paire de notes si elles sont susceptibles d'appartenir à la même couche. La distance entre deux notes est calculée entre autres d'après leur simultanéité, leur hauteur et la proximité de leur onset. Ainsi, pour chaque note, une liste des k plus proches voisins est établie. Puis les notes sont regroupées en *clusters*. Un stream mélodique peut être découpé en plusieurs petites sections, puisque la diversité des mélodies n'assure

pas toujours la cohérence à l'intérieur d'un cluster : travailler sur de larges couches les "applatit" toutes. Même si cette approche produit de bons résultats en termes de segmentation, de nombreuses couches sont encore trop éclatées pour être détectées comme des couches complètes de mélodie ou d'accompagnement. Néanmoins, des algorithmes de classification pourraient étiqueter certaines de ces couches en tant que mélodies ou accompagnements, voire détecter le type de l'accompagnement.

La seconde idée, que nous développons ici, est de *détecter directement les couches importantes à partir des données polyphoniques*. Ici, nous choisissons de nous concentrer sur les relations significatives basées sur les critères homorythmiques. Les paragraphes suivants définissent la notion de *couches synchronisées*, c'est-à-dire de suites de notes liées par une relation d'homorythmie (h/p/o/u), et montrent comment les calculer.

Couches synchronisées

Une *note* n est représentée par un triplet $(n.pitch, n.start, n.end)$, où $n.pitch$ appartient à une échelle de hauteur (qui peut être définie diatoniquement ou par demi-tons), et $n.debut$ et $n.end$ sont deux positions avec $n.start < n.end$. Deux notes n et m sont *synchronisées* (noté par $n \equiv_h m$) si elles ont le même début et la même fin.

Une *couche synchronisée* est un ensemble de deux suites de notes synchronisées consécutives (en d'autres termes, ces suites correspondent à deux voix en homorythmie). Formellement, deux suites de notes $n_1, n_2 ... n_k$ et $m_1, m_2 ... m_k$ forment une couche synchronisée quand :

— pour tout i dans $\{1, \ldots, k\}$, $\quad n_i.start = m_i.start$

— pour tout i dans $\{1, \ldots, k\}$, $\quad n_i.end = m_i.end$

— pour tout i dans $\{1, \ldots, k-1\}$, $\quad n_i.end = n_{i+1}.start$

Cette définition peut être étendue pour n'importe quel nombre de voix. Une relation p/o/u ayant une signification musicale forte, nous voulons pouvoir les connaître. On peut donc restreindre la relation $\equiv_h$ en considérant les informations de hauteur :

— on note $n \equiv_\delta m$ si l'intervalle entre les deux notes n et m est δ. La nature de l'intervalle δ dépend du modèle choisi pour les hauteurs : par exemple l'intervalle peut être diatonique, comme pour une "tierce" (mineure ou majeure), ou une approximation de l'information en demi-tons, comme pour "3 ou 4 demi-tons". Certaines couches synchronisées avec des relations $\equiv_\delta$ correspondent à des mouvements parallèles ;

— on note $n \equiv_o m$ si les notes n et m sont séparées par un nombre quelconque d'octaves ;

— on note $n \equiv_u m$ quand il y a une égalité exacte des hauteurs (unisson).

Soit une relation $\equiv \in \{\equiv_h, \equiv_\delta, \equiv_o, \equiv_u\}$, on dit qu'une couche synchronisée respecte la relation $\equiv$ si ses notes sont liées deux à deux par cette relation. La relation $\equiv_h$ est une relation d'équivalence, mais les relations restreintes n'ont pas besoin de l'être : certaines relations $\equiv_\delta$ ne sont pas transitives.

Par exemple, dans la figure 9.8, il y a, entre les voix S et A (correspondant aux violons I et II), dans les deux premières mesures :

— une couche synchronisée ($\equiv_h$) sur les deux mesures ;

— et une couche synchronisée ($\equiv_{\text{third}}$) sur les deux mesures, sauf la première note.

Remarquons que ceci ne correspond pas exactement à l'analyse de référence "musicale" (mouvement parallèle sur les 4 premières mesures) à cause de certains silences et des premières notes synchronisées qui ne sont pas à la tierce.

Une couche synchronisée est *maximale* si elle n'est pas strictement incluse dans une autre couche synchronisée. Notons que deux couches synchronisées maximales peuvent se chevaucher si elles ne sont pas synchronisées. De plus, le nombre de couches synchronisées peut croître exponentiellement avec le nombre de notes.

Détection des couches synchronisées

Un effet de texture très remarquable se produit lorsque *toutes* les voix utilisent la même texture au même moment. Par exemple, un unisson soudain frappe l'attention de l'auditeur. On peut d'abord vérifier si toutes les notes d'un segment de la partition appartiennent à une unique couche synchronisée (sous une certaine relation). Les doublures à l'octave ou à l'unisson sont prises en compte uniquement lorsqu'elles durent au moins deux temps.

Dans le cas général, la texture a plusieurs couches, et l'objectif est donc d'extraire les couches en utilisant *certaines* des notes. Rappelons que nous travaillons sur des fichiers où la polyphonie n'est pas séparée en voix. Nous voulons extraire les couches synchronisées maximales. Cependant, comme leur nombre peut croître exponentiellement avec le nombre de notes, on calcule seulement les positions de début et de fin des couches synchronisées maximales.

L'idée de notre algorithme est la suivante : récursivement, deux voix $n_1, \ldots, n_k$ et $m_1, \ldots, m_k$ sont synchronisées si et seulement si $n_1, \ldots, n_{k-1}$ et $m_1, \ldots, m_{k-1}$ sont synchronisées, n_k et m_k sont synchronisées et finalement $n_{k-1}.end = n_k.start$. Formellement, l'algorithme peut être décrit comme suit :

Étape 1. Calcul d'une table avec les couches synchronisées maximales à gauche On construit la table $\mathit{leftmost_start}_{\equiv}[j]$ contenant, pour chaque position de fin j, la position de début la plus à gauche d'une couche synchronisée respectant la relation $\equiv$ et terminant en j. Ceci peut être fait par programmation dynamique avec la récurrence suivante :

$$\mathit{leftmost_start}_{\equiv}[j] = \begin{cases} \min\{\mathit{leftmost_start}_{\equiv}[i] \mid i \in S_{\equiv}(j)\} \\ \quad \text{si } S_{\equiv}(j) \text{ n'est pas vide} \\ j \quad \text{si } S_{\equiv}(j) \text{ est vide} \end{cases}$$

où $S_{\equiv}(j)$ est l'ensemble des positions de début des notes synchronisées terminant en j et respectant la relation $\equiv$:

$$S_{\equiv}(j) = \left\{ n.start \;\middle|\; \begin{array}{l} \text{il y a deux notes différentes } n \equiv m \\ \text{telles que } n.end = j \end{array} \right\}$$

Étape 2. Sortie des couches synchronisées maximales seulement (à gauche et à droite) Donne (i, j) où $i = \mathit{leftmost_start}_{\equiv}[j]$ pour tout j tel que $j = \max\{j_o \mid \mathit{leftmost_start}_{\equiv}[j_o] = \mathit{leftmost_start}_{\equiv}[j]\}$

La première étape est réalisée en temps $O(nk)$, où n est le nombre de notes et $k \leq n$ le nombre maximal de notes sonnant simultanément, donc en temps $O(n^2)$. La deuxième étape est réalisée en temps $O(n)$ en parcourant de la droite vers la gauche la table $\mathit{leftmost_start}_{\equiv}$, et en sortant les valeurs i lorsqu'on les trouve pour la première fois.

Pour réellement trouver les intervalles, on peut stocker dans la table $\mathit{leftmost_start}_{\equiv}[j]$ une paire (i, ℓ), où ℓ est la liste des notes/intervalles à partir desquels l'ensemble des couches synchronisées peut être construit (cet ensemble peut être très grand, mais pas ℓ). La complexité en temps est alors $O(n(k+w))$, où $w \leq n$ est la plus grande taille possible de ℓ. Ainsi la complexité en temps est

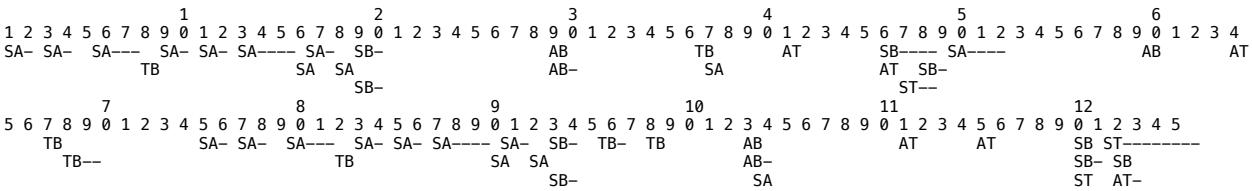


FIGURE 9.9 – Résultat de la détection de mouvement parallèle sur le premier mouvement du quatuor à cordes K. 157 n° 4 de Mozart. Les lignes du haut donnent les numéros de mesure. L'algorithme détecte 52 couches synchronisées respectant la relation $\equiv_p$. 39 de ces 52 couches chevauchent des couches identifiées dans l'analyse de référence avec des relations p/o/u. Les mouvements parallèles sont identifiés par les voix impliquées (S/A/T/B), mais cette information n'est pas utilisée dans l'algorithme qui travaille sur de la polyphonie dont les voix ne sont pas séparées.

encore en $O(n^2)$. Ceci permet, dans un second temps, de filtrer les couches synchronisées candidates en fonction de critères additionnels sur ℓ.

Notons finalement que la définition de couche synchronisée peut être étendue pour inclure les notes consécutives séparées par des silences. Le même algorithme s'applique, mais la valeur de k atteint le nombre maximum de notes qui peuvent être reliées de cette façon.

9.2.4 Résultats et discussion

Nous avons testé l'algorithme proposé pour chercher les couches synchronisées respectant la relation $\equiv_\delta$ (intervalle de hauteur constant, incluant le mouvement parallèle) sur les dix pièces de notre corpus sous forme de fichiers Humdrum `.krn` [44]. Bien que les pièces soient des quatuors à cordes, nous les considérons comme des données polyphoniques non séparées, donnant en entrée de l'algorithme un seul ensemble de notes. L'algorithme trouve 434 couches. La figure 9.9 montre un exemple de sortie de l'algorithme. Globalement, sur le corpus, l'algorithme étiquette 797 mesures (soit 53.6% de la longueur) comme couches synchronisées.

Évaluation par rapport à l'analyse de référence. Il y a dans l'analyse de référence 354 couches avec des relations p/o/u : principalement des mouvements parallèles, et quelques doublement à l'octave et unissons. Comme discuté précédemment, ces couches reportées dans l'analyse de référence correspondent à une interprétation musicale : elle ne sont pas aussi formalisées que notre définition de couche synchronisée. De plus, l'algorithme fournit moins d'informations que l'analyse de référence : quand un mouvement parallèle est trouvé, l'algorithme ne peut pas préciser à quelle voix ou à quel instrument il apparaît, puisque les voix de nos données ne sont pas séparées.

Néanmoins, nous comparons les couches prédites par l'algorithme avec celles de l'analyse de référence. Les résultats sont résumés dans la table 9.2. Une couche calculée est marquée comme vrai positif (TP) dès qu'elle chevauche une couche p/o/u de l'analyse de référence.

356 des 434 couches synchronisées calculées chevauchent les couches p/o/u de l'analyse de référence, ainsi 82% des couches synchronisées calculées sont (au moins partiellement) pertinentes musicalement. Ces 356 couches correpondent à 194 couches p/o/u de l'analyse de référence (sur 340, ce qui donne une sensibilité de 58%) : une majorité des mouvements parallèles décrits dans l'analyse de référence sont trouvées par l'algorithme.

	long. résultat	résultat	TP	FP	chevauch.	exact
Haydn Op. 33 n. 1	40m (44%)	37	32 (86.5%)	5	14 / 22 (63.6%)	7 / 22
Haydn Op. 33 n. 2	21m (22%)	17	15 (88.2%)	2	7 / 13 (53.9%)	7 / 13
Haydn Op. 33 n. 3	73m (42%)	48	44 (91.7%)	4	27 / 51 (52.9%)	15 / 51
Haydn Op. 33 n. 4	19m (21%)	47	17 (36.2%)	30	5 / 6 (83.3%)	3 / 6
Haydn Op. 33 n. 5	235m (77%)	58	47 (81.0%)	11	27 / 51 (52.9%)	11 / 51
Haydn Op. 33 n. 6	63m (37%)	24	21 (87.5%)	3	19 / 44 (43.2%)	11 / 44
Mozart K. 80 n. 1	45m (67%)	27	26 (96.3%)	1	20 / 36 (55.6%)	14 / 36
Mozart K. 155 n. 2	76m (64%)	46	44 (95.7%)	2	27 / 37 (73.0%)	15 / 37
Mozart K. 157 n. 4	62m (49%)	52	39 (75.0%)	13	15 / 24 (62.5%)	8 / 24
Schubert Op. 125 n. 1	163m (64%)	78	71 (91.0%)	7	33 / 56 (58.9%)	20 / 56
	797m (54%)	434	356 (82.0%)	78	194 / 340 (57.1%)	111 / 340

TABLE 9.2 – Évaluation de l'algorithme sur les 10 quatuors à cordes de notre corpus. Les colonnes TP et FP montrent respectivement le nombre de vrais et de faux positifs, en comparant les mouvements parallèles calculés avec l'analyse de référence. La colonne "chevauch." montre le nombre de vrais mouvements parallèles coïncidant chevauchant l'analyse de référence. La colonne "exact" restreint ces mouvements parallèles coïncidant pour lesquels les bords coïncident à ceux de l'analyse de référence (tolérance : deux temps).

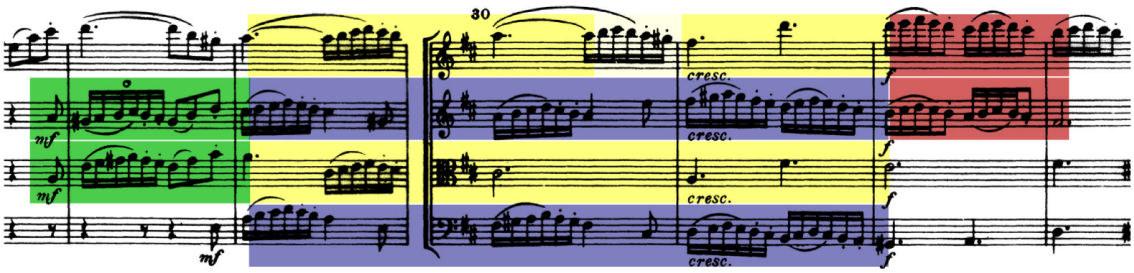


FIGURE 9.10 – Haydn, Op. 33 n° 6, m. 28-33. L'analyse de référence contient 4 mouvements parallèles.

Mouvements parallèles fusionnés Si l'on se restreint aux couches où les bords coïncident avec ceux de l'analyse de référence (même début même fin, avec une tolérence de 2 temps), le nombre de couches correctes tombe de 194 à 117. Ceci est dû au fait que l'algorithme fusionne souvent les mouvements parallèles consécutifs. Un exemple de ce défaut est décrit sur la figure 9.10. Ici une mélodie est jouée en imitation, résultant en des mouvements parallèles impliquant toutes les voix tour à tour. L'algorithme détecte une unique couche synchronisée, qui correspond à une perception globale mais donne moins d'information sur la texture. Il faut se rappeler ici que l'algorithme calcule les bords des couches synchronisées et non de véritables instances de ces couches, ce qui nécessiterait une sorte de séparation de voix et pourrait engendrer un grand nombre d'instances.

Faux positifs. Seuls 78 faux positifs sont trouvés par l'algorithme. De nombreux faux positifs (par rapport à l'analyse de référence) sont des mouvements parallèles détectés à l'intérieur d'une homorythmie (relation $\equiv_h$) entre 3 ou 4 voix. En particulier, l'algorithme détecte un mouvement parallèle dès qu'il y a des suites de notes répétées dans au moins deux voix. C'est le cas dans l'Op. 33 n° 4 de J. Haydn qui contient de nombreuses homorythmies en notes répétées, pour lesquelles

on obtient 30 faux positifs. Même en se concentrant sur les couches avec un réel “mouvement”, les faux positifs pourraient également apparaître entre une troisième voix et deux voix avec des notes répétées. Davantage de recherches sont nécessaires pour trouver un moyen d'éliminer ces faux positifs soit dans l'algorithme, soit dans une étape de filtrage ultérieure.

Ce travail est encore en cours de développement et fait partie des éléments que nous considérons comme déterminants pour obtenir une analyse de structure pertinente même dans le cas de pièces musicales complexes, comme la forme sonate. Le chapitre suivant présente maintenant nos perspectives d'un point de vue général.

Chapitre 10

Perspectives

Dans la deuxième partie de ce mémoire, j'ai présenté mes travaux sur l'analyse automatique de la structure musicale, détaillant notamment les résultats obtenus sur l'analyse de fugues, des pistes pour l'analyse de la forme sonate, ainsi que des méthodes pour prendre en compte le rythme et la texture à partir de données polyphoniques.

L'une de mes perspectives de recherche à court et moyen terme est de trouver des méthodes pour analyser automatiquement des formes complexes telles que la forme sonate.

Pour progresser vers une analyse automatisée des formes sonates, plusieurs pistes nous semblent intéressantes :

— Tout d'abord, on pourrait développer des méthodes *rassemblant plusieurs éléments analytiques* isolés (motifs, tonalités et cadences, marches harmoniques...) dans une analyse globalisante, que ce soit pour le développement ou pour la forme dans son ensemble.

— De plus, ces méthodes devraient aussi pouvoir traiter directement du *matériau polyphonique* afin de permettre l'étude des pièces pour piano.

— Un objectif serait d'identifier précisément (lorsque c'est possible) la fin du premier thème et le début du second, en s'appuyant sur la *césure médiane* (lorsqu'elle existe) [39, 40].

Pour y parvenir, une meilleure analyse de la texture semble une piste particulièrement prometteuse, combinée à d'autres éléments comme les motifs et l'harmonie. La première modélisation de la texture présentée dans ce mémoire propose un modèle en couches : il s'agit d'abord de caractériser les passages selon qu'ils sont mélodiques ou d'accompagnement. À un niveau plus fin, nous avons proposé de qualifier chaque couche selon différents agencements, parmi lesquels le mouvement parallèle (lorsque plusieurs voix sont à l'unisson ou jouent en même temps des notes séparées par une tierce, une sixte ou un octave), l'imitation (lorsque des voix jouent la même chose avec un décalage de temps), l'homophonie (lorsque des voix jouent des notes en même temps), la polyphonie (lorsque les notes jouées par les différentes voix ne sont pas nécessairement simultanées)... Pour améliorer cette première modélisation, j'ai commencé à travailler en collaboration avec deux musicologues : Marc Rigaudière (IREMus, Université Paris Sorbonne) et Florence Doé de Maindreville (Université de Reims Champagne Ardennes). Pour l'instant nous avons considéré uniquement des quatuors à cordes de la période classique, dont la complexité est appréhendable par l'ordinateur. Nous souhaitons étendre notre corpus à des quatuors plus récents (période romantique) tout en restant sur un répertoire tonal. Une fois que notre modélisation sera suffisamment fine et robuste, nous pourrons

concevoir des algorithmes permettant de détecter les textures proposées. Les moyens d'y parvenir sont en cours de discussion au sein d'Algomus, notamment avec M. Giraud et R. Groult. La caractérisation mélodie/accompagnement est une tâche ardue, d'autant plus qu'il est difficile de proposer une définition d'une mélodie qui fasse consensus. Parmi les approches envisagées, rappelons que Nicolas Guiomard-Kagan a travaillé sur les algorithmes de séparation en voix et en streams (segments en général plus courts que des voix complètes formant des groupes de notes cohérents, pas nécessairement monophoniques, contrairement aux voix). Une piste intéressante serait de les adapter et de les paramétrer afin de permettre la détection de mélodie et d'accompagnement.

Une autre perspective qui semble prometteuse pour l'analyse de structure est l'étude des progressions harmoniques et des formes cadentielles.

Les cadences sont des progressions harmoniques particulières marquant les fins de phrases musicales et les transitions entre les parties d'une pièce, et contribuant à l'impression que la musique "se termine", est "suspendue", ou "part dans d'autres directions". Elles sont le principal moyen d'articuler le but tonal d'une pièce, autant pour des formes thématiques simples que pour des œuvres ayant une construction formelle à grande échelle comme la forme sonate (voir la figure 10.1).

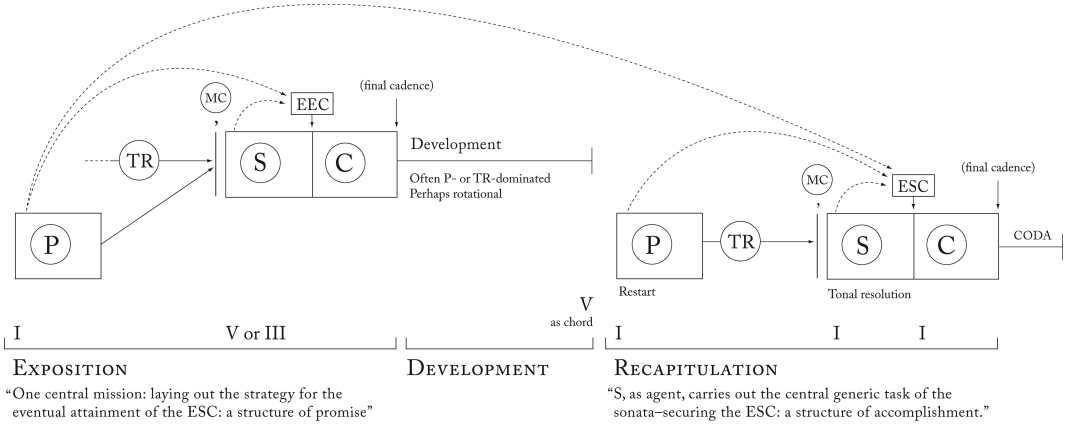


Figure 10.1 – La forme sonate est structurée par des thèmes (en général deux, un thème principal et un thème secondaire), mais aussi par un plan tonal : le thème principal est exposé dans la tonalité principale ou tonique, et une transition amène le thème secondaire à la dominante (c'est-à-dire 5 tons plus haut), et dans la réexposition ces deux thèmes sont repris à la tonique. Les cadences (MC : Medial Caesura, EEC : Essential Expositional Closure/ESC : Essential Structural Closure, final cadence) marquent les transitions entre les différentes sections. Schéma extrait de [40].

Modéliser les cadences est un problème complexe qui est au cœur des préoccupations actuelles des musicologues [6]. En effet, bien que dans les ouvrages pédagogiques sur l'harmonie, les définitions des principaux types de cadences soient plutôt sommaires, les musicologues s'accordent sur la complexité de définir les cadences, de par la variété des réalisations cadentielles observées dans le répertoire classique. Pour étudier les cadences, il est donc nécessaire de faire une étude approfondie de différents corpus. Dans [6], N. Martin et J. Pedneault-Deslauriers ont ainsi étudié les demi-cadences dans les sonates pour piano de Mozart. Mais l'analyse manuelle, longue et complexe, rend difficile l'analyse systématique de corpus d'envergure, qui permettrait d'importantes avancées sur la compréhension de l'évolution des choix compositionnels au cours du temps. Dans le même

ouvrage, Rohrmeier et Neuwirth proposent une première caractérisation partielle en utilisant des grammaires, basée sur les degrés et la ligne de basse.

Très peu d'études informatiques se sont intéressées à l'identification des progressions harmoniques [77] ou à leur représentation sous forme de trajectoires musicales [8]. [25] s'inspire des travaux de Rohrmeier pour créer un système permettant de dériver les relations harmoniques entre les accords. Les règles de la grammaire sont inférées et testées pour des harmonies jazz. À l'heure actuelle, seuls quelques algorithmes comme [GGLL2015] ou [45] permettent de reconnaître les cadences les plus simples. Les outils des musicologues pour traiter des progressions plus complexes nécessitent une part de traitement manuel non négligeable [36]. À notre connaissance, il n'y a pas d'étude algorithmique proposant une modélisation complète et fonctionnelle des progressions harmoniques, intégrant la reconnaissance des cadences et de leur préparation.

Pour initier ce projet, je collabore avec M. Giraud, L. Bigo (CRIStAL, Univ. Lille), C. Guillotel-Nothmann (IREMus, Univ Paris Sorbonne) et Richard Groult (MIS, UPJV). Les cadences que nous cherchons à modéliser et à détecter en priorité seront les cadences parfaites, ainsi que leur préparation, et les demi-cadences, susceptibles d'apporter de grandes avancées dans la détection globale de forme. Pour déduire les cadences comme les progressions harmoniques à partir des éléments contextuels, nous privilégions l'utilisation d'algorithmes d'apprentissage. L'une des difficultés de l'utilisation des algorithmes d'apprentissage est que les résultats dépendent du corpus utilisé. Nous continuons de collaborer avec des musiciens et des musicologues pour définir les progressions harmoniques les plus pertinentes et pour construire le corpus d'évaluation de nos algorithmes.

L'avancée sur ces deux problématiques devrait permettre une meilleure analyse de la forme musicale par la meilleure détection des fins de parties et des transitions entre les parties.

Enfin, comme nous l'avons détaillé dans [GGL2016], l'un des enjeux majeurs pour évaluer et étalonner nos méthodes est de réaliser des *fichiers d'analyse de référence* tels que ceux réalisés sur les fugues [GGLL2015]. Ces fichiers devraient s'appuyer sur des analyses musicologiques reconnues, comme par exemple, pour les oeuvres de Beethoven, celles de Helm, Radcliffe ou Tovey [38, 83, 103]. Ils contiendraient, pour un corpus varié, la position et la longueur des différents thèmes dans l'exposition et la réexposition, des motifs dans le développement, les cadences et le plan tonal global de chaque pièce. On devrait aussi formaliser dans ces fichiers les ambiguïtés possibles d'analyse [10]. Pour atteindre cet objectif, la collaboration constante avec des musicologues est indispensable.

Liste de mes publications

L'ensemble des publications est présenté par ordre alphabétique des auteurs, comme c'est l'usage en combinatoire des mots et en systèmes distribués, sauf les publications liées à la thèse de Nicolas Guiomard-Kagan où il est présenté comme premier auteur suivi des autres auteurs par ordre alphabétique.

Chapitre de livre (1)

[GGL2016] M. Giraud, R. Groult, F. Levé, *Computational Analysis of Musical Form*, ed. David Meredith, in *Computational Music Analysis*, Springer, isbn 978-3-319-25929-1, p. 113-136, 2016.

Articles de journaux internationaux avec comité de lecture (10)

[GL2015] A. Glen, F. Levé, *Generalized trapezoidal words*, Journal of Combinatorial Theory, Series A, Volume 136, p96-125, 2015.

[GGLL2015] M. Giraud, R. Groult, E. Leguy, F. Levé, *Computational Fugue Analysis*, Computer Music Journal, Volume 39 (2), p77-96, 2015.

[DLPV2013] Y. Dieudonné, F. Levé, F. Petit, V. Villain, *Deterministic geoleader election in disoriented anonymous systems*, Theoretical Computer Science, Volume 506, p43-54, 2013.

[GLR2009] A. Glen, F. Levé, G. Richomme, *Directive words of episturmian words : equivalences and normalization*, RAIRO-Theoretical Informatics and Applications, Volume 43 (2), p299-319, 2009.

[GLR2008b] A. Glen, F. Levé, G. Richomme, *Quasiperiodic and Lyndon episturmian words*, Theoretical Computer Science 409(3), p578-600, 2008.

[LR2007b] F. Levé, G. Richomme, *Quasiperiodic Sturmian words and morphisms*, Theoretical Computer Science 372(1), p15-25, 2007.

[LR2005] F. Levé, G. Richomme, *On a conjecture about finite fixed points of morphisms*, Theoretical Computer Science, Volume 339 (1), p103-128, 2005.

[LR2004] F. Levé, G. Richomme, *Quasiperiodic infinite words : some anwers*, Bulletin of the European Association for Theoretical Computer Science 84, p128-138, 2004.

[LS2003] F. Levé, P. Séébold, Conjugation of standard morphisms and a generalization of singular words, Bull. Belg. Math. Soc. 10, p737-748, 2003.

[LS2001] F. Levé, P. Séébold, Proof of a conjecture on word complexity, Bull. Belg. Math. Soc. 8, p277-291, 2001.

Articles de conférences internationales avec comité de lecture (14)

[LMV2016] F. Levé, K. Mohamed, and V. Villain, *Snap-Stabilizing PIF on Arbitrary Connected Networks In Message Passing Model*, accepted in SSS'16 (18th International Symposium on Stabilization, Safety and Security of Distributed Systems), Nov 7-10, 2016, Lyon, France.

[GGGL2016] N. Guiomard-Kagan, M. Giraud, R. Groult, F. Levé, *Improving voice separation by better connecting contigs*, Proceedings of ISMIR'16 (17th international society for music information retrieval conference).

[GGGL2015] N. Guiomard-Kagan, M. Giraud, R. Groult, F. Levé, *A comparison of voice and stream segmentation algorithms*, Proceedings of ISMIR'15 (16th international society for music information retrieval conference). (Taux d'acceptation : 47.1%, 114 articles acceptés sur 242 soumissions).

[GLMRT2014] M. Giraud, F. Levé, F. Mercier, M. Rigaudière, D. Thorez, *Modeling Texture in Symbolic Musical Data*, Proceedings of ISMIR'14 (15th international society for music information retrieval conference), Oct 27-31, 2014, Taipei, Taiwan.

[LMV2014] F. Levé, K. Mohamed, and V. Villain, *Snap-Stabilizing PIF on Non-Oriented Trees and Message Passing Model*, Proceedings of SSS'14 (16th International Symposium on Stabilization, Safety, and Security of Distributed Systems), Sept 28 - Oct 1, 2014, Padeborn, Germany. (Taux d'acceptation : 47.7%)

[LR2013] F. Levé, G. Richomme, *On quasiperiodic morphisms*, WORDS 2013 Conference, september 16-20 2013 Turku, Finland. Lecture Notes in Computer Science Volume 8079, p. 181-192, 2013.

[GGL2012b] M. Giraud, R. Groult, F. Levé, *Detecting episodes with Harmonic Sequences for Fugue analysis*, Proceedings of ISMIR'12 (13th international society for music information retrieval conference), october 8-12, 2012, Porto, Portugal. (Taux d'acceptation : 44%, 95 articles acceptés sur 215 soumissions).

[GGL2012a] M. Giraud, R. Groult, F. Levé, *Subject and counter-subject detection for analysis of the Well-Tempered Clavier fugues*, 9th International Symposium on Computer Music Modeling and Retrieval (CMMR) "Music and Emotions", june 19-22, 2012, Queen Mary University of London, UK.

[AGGGLS2011] G. Arnaud, R. Gaymay, M. Giraud, R. Groult, F. Levé, C. Séguin, *Rhythm extraction from polyphonic symbolic music*, proceedings of ISMIR'11 (12th international society for music information retrieval conference), october 24-28, 2011, Miami, Florida (USA), p. 375-380.

[GLR2008a] A. Glen, F. Levé, G. Richomme, *Directive words of episturmian words*, 12ièmes Journées Montoises d'Informatique Théorique, août 2008, Mons (Belgique).

[LR2007a] F. Levé, G. Richomme, *Quasiperiodic episturmian words*, proceedings of WORDS'07 6th International Conference on Combinatorics on Words, Marseille (France).

[LR2003] F. Levé, G. Richomme, *On a conjecture about finite fixed points of morphisms (extended abstract)*, proceedings of WORDS'03 4th International Conference on Combinatorics on Words, Turku (Finlande), Turku Centre for Computer Science General Publication No 27 (August 2003), p198-206.

[LS2002] F. Levé, P. Séébold, *Conjugation of standard morphisms and a generalization of singular words*, 9ièmes Journées Montoises d'Informatique Théorique, septembre 2002, Montpellier.

[LS2000] F. Levé, P. Séébold, *Preuve d'une conjecture sur la complexité des mots*, 8ièmes Journées Montoises d'Informatique Théorique, mars 2000, Marne-la-Vallée.

Articles de conférences nationales avec comité de lecture (2)

[GGGL2014] Nicolas Guiomard-Kagan, Mathieu Giraud, Richard Groult, Florence Levé. *Séparation de voix en musique classique et en musique populaire.* Journées d'Analyse Musicale (JAM 2014), 15-16 décembre 2014, Paris, France.

[DGGLL2014] Laurent David, Mathieu Giraud, Richard Groult, Florence Levé, Corentin Louboutin. *Vers une analyse automatique des formes sonates.* Journées d'Informatique Musicale (JIM 2014), 21-23 Mai 2014, Bourges, France. Journées d'Informatique Musicale 2014, pp. 113-118.

Autres communications

[GGLL2014] Mathieu Giraud, Richard Groult, Emmanuel Leguy, Florence Levé. *Analyse automatique de la structure des fugues.* Journée du GDR Esthétique, Arts & Sciences. 15 avril 2014, Lille.

Bibliographie

[1] J.-P. Allouche and T. Johnson. Finite automata and morphisms in assisted musical composition. *Journal of New Music Research*, 24, 1995.

[2] M. Andreatta and M. Chemillier. Modèles mathématiques pour l'informatique musicale (MMIM) : outils théoriques et stratégies pédagogiques. In *Journées d'Informatique Musicale*, pages 113–123, Lyon, avril 2007.

[3] A. Apostolico, M. Farach, and C. S. Iliopoulos. Optimal superprimitivity testing for strings. *Information Processing Letters*, 39(1) :17–20, 1991.

[4] P. Arnoux and G. Rauzy. Représentation géométrique de suites de complexités $2n + 1$. *Bull. Soc. Math. France*, 119 :199–215, 1991.

[5] Anthony Bergerault. L'enseignement du contrepoint et de la fugue au Conservatoire de Paris (1858-1905). *Transposition. Musique et sciences sociales*, 1, 2011.

[6] Pieter Bergé and Markus Neuwirth, editors. *What Is a Cadence ? : Theoretical and Analytical Perspectives on Cadences in the Classical Repertoire*. Leuven University Press, 2015.

[7] V. Berthé, C. Holton, and L. Q. Zamboni. Initial powers of Sturmian sequences. *Acta Arithmetica*, 122 :315–347, 2006.

[8] Louis Bigo, Moreno Andreatta, Jean louis Giavitto, Olivier Michel, Antoine Spicher, and Lacl/université Paris est Creteil. Computation and visualization of musical structures in chord-based simplicial complexes," submited to mathematics and computation in music, 2013.

[9] Louis Bigo, Jean-Louis Giavitto, Moreno Andreatta, Olivier Michel, and Antoine Spicher. Computation and Visualization of Musical Structures in Chord-Based Simplicial Complexes. In *Mathematics and Computation in Music (MCM 2013)*, pages 38–51, 2013.

[10] Frédéric Bimbot, Emmanuel Deruty, Gabriel Sargent, and Emmanuel Vincent. Semiotic structure labeling of music pieces : Concepts, methods and annotation conventions. In *International Society for Music Information Retrieval Conference (ISMIR 2012)*, pages 235–240, 2012.

[11] Frédéric Bimbot, Emmanuel Deruty, Gabriel Sargent, and Emmanuel Vincent. System & Contrast : A Polymorphous Model of the Inner Organization of Structural Segments within Music Pieces. *Music Perception*, page 41, 2016. To appear - http ://mp.ucpress.edu/.

[12] Albert S. Bregman. *Auditory scene analysis*. Bradford, Cambridge, 1990.

[13] Siglind Bruhn. *J. S. Bach's Well-Tempered Clavier. In-depth Analysis and Interpretation*. Mainer International, 1993.

[14] Emilios Cambouropoulos. A general pitch interval representation : Theory and applications. *Journal of New Music Research*, 25(3) :231–251, 1996.

[15] Emilios Cambouropoulos. Voice separation : theoretical, perceptual and computational perspectives. In *Int. Conf. on Music Perception and Cognition (ICMPC)*, 2006.

[16] Emilios Cambouropoulos, Maxime Crochemore, Costas S. Iliopoulos, Manal Mohamed, and Marie-France Sagot. A pattern extraction algorithm for abstract melodic representations that allow partial overlapping of intervallic categories. In *International Conference on Music Information Retrieval (ISMIR 2005)*, pages 167–174, 2005.

[17] William E. Caplin. *Classical Form : A Theory of Formal Functions for the Instrumental Music of Haydn, Mozart, and Beethoven.* Oxford University Press, 2000.

[18] William E. Caplin, James Hepokoski, and James Webster. *Musical Form, Forms & Formenlehre – Three Methodological Reflections.* Leuven University Press, 2009.

[19] Marc Chemillier. *Grammaires, automates et musique*, pages 195–230. Hermès, Lavoisier, 2004.

[20] Nicholas Cook. *A guide to musical analysis.* Oxford University Press, Dent, 1987.

[21] Grosvenor Cooper and Leonard B. Meyer. *The Rhythmic Structure of Music.* University of Chicago Press, 1960.

[22] Michael Scott Cuthbert and Christopher Ariza. music21 : A toolkit for computer-aided musicology and symbolic music data. In *International Society for Music Information Retrieval Conference (ISMIR 2010)*, pages 637–642, 2010.

[23] Carl Czerny. *School of Practical Composition.* R. Cocks, London, 1848.

[24] Roger B. Dannenberg and Masataka Goto. *Handbook of Signal Processing in Acoustics*, chapter Music Structure Analysis, pages 305–331. Springer, 2008.

[25] W. Bas De Haas, José Pedro Magalhães, Frans Wiering, and Remco C. Veltkamp. HarmTrace : Automatic functional harmonic analysis. *Computer Music Journal*, 37 :4 :37–53, 2013.

[26] A. de Luca. Sturmian words : structure, combinatorics and their arithmetics. *Theoret. Comput. Sci.*, 183 :45–82, 1997.

[27] Diana Deutsch and John Feroe. The internal representation of pitch sequences in tonal music. *Psychological Review*, 88(6) :503–522, 1981.

[28] X. Droubay, J. Justin, and G. Pirillo. Episturmian words and some constructions of de Luca and Rauzy. *Theoret. Comput. Sci.*, 255(1-2) :539–553, 2001.

[29] Jonathan Mark Dunsby. Considerations of texture. *Music and letters*, 70(1) :46–57, 1989.

[30] F. Durand. A characterization of substitutive sequences using return words. *Discrete Math.*, 179 :89–101, 1998.

[31] François-Joseph Fétis. *Traité du contrepoint et de la fugue.* Troupenas, Paris, 1825.

[32] André Gedalge. *Traité de la fugue.* Enoch, Paris, 1901.

[33] Asif Ghias, Jonathan Logan, David Chamberlin, and Brian C. Smith. Query by humming : musical information retrieval in an audio database. In *ACM Multimedia*, pages 231–236, 1995.

[34] A. Glen and J. Justin. Episturmian words : a survey, preprint, 2007.

[35] A. Glen, J. Justin, and G. Pirillo. Characterizations of finite and infinite episturmian words via lexicographic orderings. *European J. Combin.*, 29 :45–58, 2008.

[36] Christophe Guillotel-Nothmann. Telos : logiciel d'analyse harmonique et contrapuntique, 2009.

[37] Pierre Hanna and Matthias Robine. Query by tapping system based on alignment algorithm. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2009)*, pages 1881–1884, 2009.

[38] Theodor Helm. *Beethovens Streichquartette : Versuch einer technischen Analyse dieser Werke im Zusammenhange mit ihren geistigen Gehalt.* Leipzig, 1885.

[39] James Hepokoski and Warren Darcy. The medial caesura and its role in the eighteenth-century sonata exposition. *Music Theory Spectrum*, 19(2) :115–154, 1997.

[40] James Hepokoski and Warren Darcy. *Elements of Sonata Theory : Norms, Types, and Deformations in the Late-Eighteenth-Century Sonata.* Oxford University Press, 2006.

[41] Yuzuru Hiraga. Structural recognition of music by pattern matching. In *Int. Computer Music Conference (ICMC 97)*, pages 426–429, 1997.

[42] C. Holton and L.Q. Zamboni. Descendants of primitive substitutions. *Theory Comput. Syst.*, 32 :133–157, 1999.

[43] David Huron. Characterizing musical textures. In *International Computer Music Conference (ICMC)*, pages 131–134, 1989.

[44] David Huron. Music information processing using the Humdrum toolkit : Concepts, examples, and lessons. *Computer Music Journal*, 26(2) :11–26, 2002.

[45] Plácido R. Illescas, David Rizo, and José M. Iñesta. Harmonic, melodic, and functional automatic analysis. In *Proceedings of the 2007 International Computer Music Conference*, volume I, pages 165–168, 2007.

[46] Jyh-Shing Jang, Hong-Ru Lee, and Chia-Hui Yeh. Query by Tapping : a new paradigm for content-based music retrieval from acoustic input. In *Advances in Multimedia Information Processing (PCM 2001), LNCS 2195*, pages 590–597, 2001.

[47] Nanzhu Jiang and Meinard Müller. Automated methods for analyzing music recordings in sonata form. In *International Society for Music Information Retrieval Conference (ISMIR 2013)*, pages 595–600, 2013.

[48] J. Justin. Episturmian morphisms and a Galois theorem on continued fractions. *Theoret. Inform. Appl.*, 39(1) :207–215, 2005.

[49] J. Justin and G. Pirillo. Episturmian words and episturmian morphisms. *Theoret. Comput. Sci.*, 276(1-2) :281–313, 2002.

[50] J. Justin and G. Pirillo. On a characteristic property of Arnoux-Rauzy sequences. *Theoret. Inform. Appl.*, 36(4) :385–388, 2003.

[51] J. Justin and G. Pirillo. Episturmian words : shifts, morphisms and numeration systems. *Internat. J. Found. Comput. Sci.*, 15(2) :329–348, 2004.

[52] Kent Kennan. *Counterpoint, 4th edition.* Prentice Hall, 1999.

[53] Anssi Klapuri and Manuel Davy. *Signal Processing Methods for Music Transcription.* Springer, 2006.

[54] Carol L. Krumhansl and Edward journal Kessler. Tracing the dynamic changes in perceived tonal organisation in a spatial representation of musical keys. *Psychological Review*, 89(2) :334–368, 1982.

[55] Jan LaRue. *Guidelines for Style Analysis.* Harmonie Park Press, 1970.

[56] Kjell Lemström and Pauli Laine. Musical information retrieval using musical parameters. In *International Computer Music Conference (ICMC 1998)*, pages 341–348, 1998.

[57] Fred Lerdahl and Ray S. Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983, 1996.

[58] Janet M. Levy. Texture as a sign in classic and early romantic music. *J. of the American Musicological Society*, 35(3) :482–531, 1982.

[59] M. Lothaire. *Algebraic Combinatorics on Words*, volume 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge Univ. Press, 2002.

[60] Soren T. Madsen and Gerhard Widmer. Key-finding with interval profiles. In *International Computer Music Conference (ICMC 2007)*, 2007.

[61] S. Marcus. Bridging two hierarchies of infinite words. *Journal of Universal Computer Science*, 8 :292–296, 2002.

[62] S. Marcus. Quasiperiodic infinite words. *Bulletin of the European Association for Theoretical Computer Science*, 82 :170–174, 2004.

[63] Adolf Bernhard Marx. *Die Lehre von der musikalischen Komposition, praktisch teoretisch*. Breitkopf und Härtel, 1837-1847.

[64] Cory McKay. *Automatic music classification with jMIR*. PhD thesis, McGill University, 2010.

[65] Cory McKay and Ichiro Fujinaga. jSymbolic : A feature extractor for MIDI files. In *Int. Computer Music Conf. (ICMC)*, pages 302–305, 2006.

[66] David Meredith. The ps13 pitch spelling algorithm. *Journal of New Music Research*, 35(2) :121–159, 2006.

[67] David Meredith, Kjell Lemström, and Geraint A Wiggins. Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4) :321–345, 2003.

[68] Marcel Mongeau and David Sankoff. Comparison of musical sequences. *Computers and the Humanities*, 24(3) :161–175, 1990.

[69] T. Monteil. Quasiperiodic words : the multiscaled case and dynamical properties. corr, abs/math/0603354, 2006.

[70] T. Monteil. *Ph.D thesis : Illumination dans les billards polygonaux et dynamique symbolique*. PhD thesis, Université de la Méditerranée, Faculté des Sciences de Luminy, December 2005.

[71] M. Morse and G.A. Hedlund. Symbolic Dynamics II : Sturmian trajectories. *Amer. J. Math.*, 61 :1–42, 1940.

[72] Jean-Jacques Nattiez. *Musicologie générale et sémiologie*. Christian Bourgeois, 1987.

[73] Oriol Nieto and Morwaread M. Farbood. Perceptual evaluation of automatically extracted musical motives. In *Proceedings of the 12th International Conference on Music Perception and Cognition*, pages 723–727, 2012.

[74] Hugo Norden. *Foundation Studies in Fugue*. Crescendo Publishing, 1977.

[75] Quentin R. Nordgren. A measure of textural patterns and strengths. *journal of Music Theory*, 4(1) :19–31, 1960.

[76] Keith S. Orpen and David Huron. Measurement of similarity in music : A quantitative approach for non-parametric representations. *Computers in Music Research*, 4 :1–44, 1992.

[77] J.F. Paiement, D. Eck, and S. Bengio. A probabilistic model for chord progressions. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 312–319, London : University of London, 2005.

[78] G. Paquin and L. Vuillon. A characterization of balanced episturmian sequences. Technical report, LAMA, 2006. see arXiv :math/0611576v2 [math.CO].

[79] Sami Perttu. Combinatorial pattern matching in musical sequences. Master Thesis, Univ. Helsinki, 2000.

[80] Walter Piston. *Orchestration*. Norton, 1955.

[81] Ebenezer Prout. *Analysis of J. S. Bach's forty-eight fugues (Das Wohltemperierte Clavier)*. E. Ashdown, 1910.

[82] N. Pytheas Fogg. *Substitutions in Dynamics, Arithmetics and Combinatorics*, volume 1794 of *Lecture Notes in Mathematics*. Springer, 2002.

[83] Phillip Radcliffe. *Beethoven's String Quartets*. E. P. Dutton, 1965.

[84] Dimitrios Rafailidis, Alexandros Nanopoulos, Yannis Manolopoulos, and Emilios Cambouropoulos. Detection of stream segments in symbolic musical data. In *Int. Conf. on Music Information Retrieval (ISMIR)*, pages 83–88, 2008.

[85] Leonard Ratner. *Classical Music : Expression, Form, and Style*. Schirmer, 1980.

[86] Erwin Ratz. *Einführung in die musikalische Formenlehre*. Universal Edition, 1973. Second edition.

[87] Anton Reicha. *Traité de haute composition musicale*. A. Diabelli, 1824.

[88] G. Richomme. Conjugacy and episturmian morphisms. *Theoretical Computer Science*, 302(1-3) :1–34, 2003.

[89] G. Richomme. A local balance property of episturmian words. In *Proc. DLT '07*, volume 4588 of *Lecture Notes in Computer Science*, pages 371–381. Springer, Berlin, 2007.

[90] Marc Rigaudière. *La théorie musicale germanique du XIXe siècle et l'idée de cohérence*. Société Française de Musicologie, 2009.

[91] R.N. Risley and L.Q Zamboni. A generalization of Sturmian sequences : combinatorial structure and transcendence. *Acta Arith.*, 95 :167–184, 2000.

[92] Matthias Robine, Thomas Rocher, and Pierre Hanna. Improvements of key-finding methods. In *International Computer Music Conference (ICMC 2008)*, 2008.

[93] Lewis Rowell. *Thinking about Music : An Introduction to the Philosophy of Music*. Univ. of Massachusetts, 1985.

[94] Nicolas Saint-Arnaud and Kris Popat. Analysis and synthesis of sound textures. In David F. Rosenthal and Hiroshi G. Okuno, editors, *Computational Auditory Scene Analysis*, pages 293–308. Erlbaum, 1998.

[95] Gabriel Sargent, Frédéric Bimbot, and Emmanuel Vincent. A regularity-constrained Viterbi algorithm and its application to the structural segmentation of songs. In *International Society for Music Information Retrieval Conference (ISMIR 2011)*, 2011.

[96] Heinrich Schenker. *Der freie Satz*. Universal Edition, 1935.

[97] Arnold Schoenberg. *Fundamentals of musical composition*. Faber & Faber, 1967.

[98] P. Séébold. Fibonacci morphisms and Sturmian words. *Theoretical Computer Science*, 88(2) :365–384, 1991.

[99] Gerda Strobl, Gerhard Eckel, and Davide Rocchesso. Sound texture modeling : a survey. In *Sound and Music Computing (SMC)*, 2006.

[100] David Temperley. What's key for key ? the Krumhansl-Schmuckler key-finding algorithm reconsidered. *Music Perception*, 17(1) :65–100, 1999.

[101] Atte Tenkanen and Fernanda Gualda. Detecting changes in musical texture. In *Int. Workshop on Machine Learning and Music*, 2008.

[102] Godfried T. Toussaint. A comparison of rhythmic similarity measures. In *Int. Society for Music Information Retrieval Conf. (ISMIR 2004)*, 2004.

[103] Donald Francis Tovey. *A Companion to Beethoven's Pianoforte Sonatas : Complete Analyses.* AMS Press (reedited in 1998), 1931.

[104] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5) :293–302, July 2002.

[105] Alexandra L. Uitdenbogerd. *Music Information Retrieval Technology.* PhD thesis, RMIT University, Melbourne, Victoria, Australia, 2002.