

Département Textes, Informatique, Multilinguisme

## Année universitaire 2020-2021



# TABLE DES MATIÈRES

<b>Liste des figures</b>	<b>4</b>
<b>Liste des tableaux</b>	<b>5</b>
<b>Introduction générale</b>	<b>7</b>
<b>1 Contexte</b>	<b>11</b>
1.1 Langues naturelles et musique en informatique . . . . .	12
1.2 La transcription automatique de la musique . . . . .	14
1.3 La transcription automatique de la batterie . . . . .	15
1.4 Les représentations de la musique . . . . .	16
<b>2 État de l'art</b>	<b>21</b>
2.1 Monophonique et polyphonique . . . . .	21
2.2 De l'enregistrement audio vers le MIDI . . . . .	22
2.3 Du format MIDI vers une partition . . . . .	23
2.4 Approche linéaire et approche hiérarchique . . . . .	23
<b>3 Méthodes</b>	<b>27</b>
3.1 La notation de la batterie . . . . .	27
3.2 La transcription manuelle . . . . .	35
3.3 Modélisation pour la transcription . . . . .	37
3.4 Analyse syntaxique pour la transcription . . . . .	39
3.5 Les <b>formes rythmiques</b> . . . . .	42
<b>4 Expérimentations</b>	<b>49</b>
4.1 Le jeu de données . . . . .	50
4.2 Analyses et transcriptions manuelles . . . . .	51
4.3 Transcription polyphonique par parsing . . . . .	55
4.4 Réécriture guidée par une forme rythmique . . . . .	55
4.5 BILAN : résultats — évaluation — discussion . . . . .	60
<b>Conclusion générale</b>	<b>63</b>
<b>Bibliographie</b>	<b>65</b>

## LISTE DES FIGURES

1.1	Les figures de notes . . . . .	16
1.2	Rapport des figures de notes . . . . .	17
1.3	Les silences . . . . .	17
1.4	Exemple d'évènements MIDI . . . . .	18
1.5	Exemple de représentation MusicXML . . . . .	19
2.1	Le modèle de Markov caché . . . . .	24
2.2	Représentation arborescente d'une grille harmonique [12] . . . .	25
3.1	Les instruments de la batterie . . . . .	28
3.2	Hauteur et têtes de notes . . . . .	28
3.3	Point et liaison . . . . .	30
3.4	Silence joué . . . . .	31
3.5	Équivalence . . . . .	32
3.6	Séparation des voix . . . . .	33
3.7	Les accents et les ghost-notes . . . . .	34
3.8	Exemple pour les accentuations et les ghost-notes . . . . .	34
3.9	Définition du fla . . . . .	34
3.10	lilypond — extraits de code . . . . .	35
3.11	lilypond — transcription manuelle . . . . .	36
3.12	Présentation de Qparse . . . . .	39
3.13	Fichier MIDI pour les tests de grammaires . . . . .	40
3.14	Exemple de grammaire . . . . .	41
3.15	signature rythmique . . . . .	44
3.16	<b>motif</b> 4-4 binaire . . . . .	45
3.17	<b>Motif</b> 4-4 jazz . . . . .	45
3.18	<b>Forme rythmique</b> 4-4 afro-latin . . . . .	46
3.19	Simplifications — arbres et notations possibles . . . . .	47
4.1	Batterie électronique . . . . .	50
4.2	Motifs et gammes . . . . .	56
4.3	Partition d'un forme rythmique en 4/4 binaire . . . . .	57
4.4	Arbre de rythme — forme rythmique . . . . .	57
4.5	Arbre de rythme — voix haute . . . . .	58
4.6	Arbre de rythme — voix basse . . . . .	58
4.7	. . . . .	59
4.8	. . . . .	59
4.9	. . . . .	59

4.10 . . . . .	60
----------------	----

## LISTE DES TABLEAUX

1.1 speechToText vs AMT . . . . .	13
3.1 Noms des instruments de la batterie . . . . .	28
3.2 Codes, identités et pitches des instruments . . . . .	37
3.3 Systèmes . . . . .	43



# INTRODUCTION GÉNÉRALE

Ce mémoire de recherche, effectué en parallèle d'un stage à l'Inria dans le cadre du master de traitement automatique des langues de l'Inalco, contient une proposition originale ainsi que diverses contributions dans le domaine de la transcription automatique de la musique. Les travaux qui seront exposés ont tous pour objectif d'améliorer **qparse**, un outil de transcription automatique de la musique, et seront axés spécifiquement sur le cas de la batterie.

Nous parlerons de transcription musicale, en suivant des méthodes communes au domaine du traitement automatique des langues (TAL) plutôt que directement de langues naturelles, et nous parlerons aussi de génération automatique de partitions de musique à partir de données audio ou symboliques. En considérant que la musique à l'instar des langues naturelles est un moyen qui nous sert à exprimer nos ressentis sur le monde et les choses, ce travail reposera sur une citation de l'ouvrage de Danhauser [1] : « La musique s'écrit et se lit aussi facilement qu'on lit et écrit les paroles que nous prononçons. » L'exercice exposé dans ce mémoire nécessitera donc la manipulation d'un langage musical qui peut être analysé à l'aide de théories formelles et d'outils adéquats comme des grammaires (solfège, durées, nuances, volumes) et soulèvera des problématiques qui peuvent être résolues par l'utilisation de méthodes issues de l'informatique et de l'analyse des langues et des langages.

## POURQUOI?

- sujet traité : la batterie
- intérêt spécifique de la génération de partition de batterie comparativement au autres instrument
- patrimoine
- rapidité de génération (musicien ou enseignement)
- ...

<flo>il faut revoir la fin, avec une description rapide du problème, de la méthode suivie et des contributions suivi d'un petit plan par parties.</flo>

## COMMENT?

→ Problématique :

L'écriture musicale offre de nombreuses possibilités pour la transcription d'un rythme donné. Le contexte musical ainsi que la lisibilité d'une

partition pour un batteur entraîné conditionnent les choix d'écriture. Reconnaître la signature rythmique principale d'un rythme, la façon de regrouper les notes par des ligatures, ou simplement décider d'un usage pour une durée parmi les différentes continuations possibles (notes pointées, liaisons, silences, etc.) constituent autant de possibilités que de difficultés <dam>que de choix de représentation à réaliser?</dam>. De plus, la batterie est dotée d'une écriture spécifique par rapport à la majorité des instruments.

→ Méthodes :

→ Contributions :

<louison>liste des contributions : donner une échelle, un point de comparaison, du contexte, pour pouvoir mesurer l'importance de chaque contribution</louison>

La proposition principale de ce mémoire est basée sur la recherche de rythmes génériques sur l'*input*. Ces rythmes sont des *patterns* standards de batterie définis au préalable et accompagnés par les différentes combinaisons qui leur sont propres. On les nomme systèmes (voir sections 3.5, ??). L'objectif des systèmes est de fixer des choix le plus tôt possible afin de simplifier le reste des calculs en éliminant une partie d'entre eux. Ces choix concernent notamment la signature rythmique et les règles de réécriture.

La proposition ci-dessus a nécessité plusieurs sous-tâches :

- une modélisation de la notation de la batterie (fusion de 3.1 et de 3.3) qui était jusqu'à présent inexistante.
- plusieurs transcriptions manuelles dans le but d'analyser les contenus des fichiers MIDI et Audio (??) et de faire des comparaisons de transcription avec des outils déjà existants <sup>1</sup>.
- une partition de référence transcrite manuellement sur l'entièreté d'une performance du jeu de données afin de repérer les éléments importants pour la modélisation et de faire les liens entre les critères des données d'*input* avec l'écriture finale (3.11). Cette partition avait aussi pour objectif d'effectuer des tests et des évaluations.
- le passage au polyphonique en théorie et en implémentation impliquant la théorie sur la détection de l'identité de notes dans un Jam <sup>2</sup> et l'implémentation de tests unitaires sur le traitement des Jams (??).
- la création de grammaires pondérées spécifiques à la batterie (??)

---

1. MuseScore3

2. groupe de notes rassemblées en raison d'un faible écart entre leur emplacements temporels



L'ensemble de ces sous-tâches a permis deux réalisations principales :

1) Obtenir des arbres de rythmes corrects en *output* de qparse avec des exemples courts proches de la partition de référence.

2) La création d'une expérimentation théorique d'un système ?? dont le but premier est de démontrer qu'elle est implémentable et applicable à d'autres type de rythmes et dont le second objectif est de donner une méthode de création d'un système à partir d'une partition.

Ces deux réalisations recouvrent une partie du chemin à parcourir puisque pour effectuer des évaluations conséquentes sur résultat, la chaîne de traitement doit être finie afin de pouvoir vérifier de manière empirique que les systèmes, qui constituent ma contribution principale pour ce mémoire, ont permis d'améliorer qparse pour la transcription automatique de la batterie.

## PLAN

Nous présenterons le contexte (chapitre 1) suivi d'un état de l'art (chapitre 2) et nous définirons de manière générale le processus de transcription automatique de la musique pour enfin étayer les méthodes (chapitre 3) utilisées pour la transcription automatique de la batterie. Nous décrirons ensuite le corpus ainsi que les différentes expérimentations menées (chapitre 4). Nous concluerons par une discussion sur les résultats obtenus et les pistes d'améliorations futures à explorer. Les contributions apportées à l'outil qparse seront exposées dans les chapitres 3 et 4.



## CONTEXTE

### Sommaire

1.1	Langues naturelles et musique en informatique . . . . .	12
1.2	La transcription automatique de la musique . . . . .	14
1.3	La transcription automatique de la batterie . . . . .	15
1.4	Les représentations de la musique . . . . .	16

### Introduction

La transcription automatique de la musique (TAM) est un défi ancien [2] et difficile qui n'est toujours pas résolu de manière satisfaisante par les systèmes actuels. Il a engendré une grande variété de sous-tâches qui ont donné naissance au domaine de la recherche d'information musicale (RIM)<sup>1</sup>. Actuellement, en raison de la nature séquentielle et symbolique des données musicales et du fait que les travaux en TAL sont assez avancés en analyse de données séquentielles ainsi qu'en traitement du signal, de nombreux travaux de RIM font appel au TAL. Certains de ces travaux se concentrent notamment sur l'analyse des paroles de chansons<sup>2</sup>.  
<moi>Mais d'autres traitent directement la musique + ref.</moi>

Dans ce chapitre, nous parlerons de l'informatique musicale, nous montrerons les liens existants entre le RIM et le TAL ainsi qu'entre les notions de langage musical et langue naturelle. Nous traiterons également du problème de l'AMT et de ses applications.

Enfin, nous décrirons les représentations de la musique qui sont nécessaires à la compréhension du présent travail.

---

1. <https://ismir.net/>

2. NLP4MuSA, the 2nd Workshop on Natural Language Processing for Music and Spoken Audio, co-located with ISMIR 2021.

## 1.1 Langues naturelles et musique en informatique

### COMPUTER MUSIC

L'informatique musicale ou *Computer Music* regroupe l'ensemble des méthodes permettant de créer ou d'analyser des données musicales à l'aide d'outils informatiques [3]. Ce domaine implique l'utilisation de méthodes numériques pour l'analyse et la synthèse de musique<sup>3</sup>, qu'il s'agisse d'informations audio, ou symboliques (aide à l'écriture, transcription, base de partitions...). Un exemple de tâche dans ce domaine pourrait être l'analyse de la structure de la musique et de la reconnaissance des accords<sup>4</sup>.

### RIM

La RIM est née du domaine de l'informatique musicale et apparaît vers le début des années 2000 [5]. L'objectif de cette science est la recherche et l'extraction d'informations à partir de données musicales. Il s'agit d'un vaste champ de recherche pluridisciplinaire, à l'intersection de acoustique, signal, synthèse sonore, informatique, sciences cognitives, neurosciences, musicologie, psycho-acoustique, etc. Cette discipline récente a notamment été soutenue par de grandes entreprises technologiques<sup>5 6 7</sup> qui veulent développer des systèmes de recommandation de musique ou des moteurs de recherche dédiés au son et à la musique.

### RIM et TAL

Aborder la musique comme un langage avec des méthodes de TAL nécessite une réflexion autour de la musique en tant que langage ainsi que la possibilité de comparer ce même langage avec les langues naturelles. Léonard Bernstein [6] a donné une série de six conférences publiques à Harvard fondées en grande partie sur les théories linguistiques que Noam Chomsky a exposées dans son livre « Language and Mind ». Lors de la première conférence, qui a eu lieu le 9 octobre 1973, Bernstein a avoué être hanté par la notion d'une grammaire musicale mondiale innée et il analyse dans ses trois premières conférences, la musique en termes linguistiques (phonologie, syntaxe et sémantique). Quelques travaux en neurosciences ont également abordé ces questions, notamment par observation des processus cognitifs et neuronaux que les systèmes de trai-

---

3. Voir la transformée de Fourier pour la musique dans [4]

4. En musique, un accord est un ensemble de notes considéré comme formant un tout du point de vue de l'harmonie. Le plus souvent, ces notes sont jouées simultanément; mais les accords peuvent aussi s'exprimer par des notes successive

5. <https://research.deezer.com/>

6. <https://magenta.tensorflow.org/>

7. <https://research.atspotify.com/>

tement de ces deux productions humaines avaient en commun. Dans le travail de Poulin-Charronnat *et al.* [7], la musique est reconnue comme étant un système complexe spécifique à l'être humain dont une des similitudes avec les langues naturelles est l'émergence de régularités reconnues implicitement par le système cognitif. La question de la pertinence de l'analogie entre langues naturelles et langage musical a également été soulevée à l'occasion de projets de recherche en TAL. Keller *et al.* [8] ont exploré le potentiel de ces techniques à travers les plongements de mots et le mécanisme d'attention pour la modélisation de données musicales. La question de la sémantique d'une phrase musicale apparaît, selon eux, à la fois comme une limite et un défi majeur pour l'étude de cette analogie. Ces considérations nous rapproche de la sémiologie de F. de Saussure en tant que science générale des signes et dont la langue ne serait qu'un cas particulier, caractérisé par l'arbitrariété totale de ses unités [9].

D'autres travaux très récents, ont aussi été révélés lors de la *première conférence sur le NLP pour la musique et l'audio (NLP4MusA 2020)*. Lors de cette conférence, Jiang *et al.* [10] ont présenté leur implémentation d'un modèle de langage musical visant à améliorer le mécanisme d'attention par élément, déjà très largement utilisé dans les modèles de séquence modernes pour le texte et la musique.

Le domaine du TAL qui se rapproche le plus du RIM est la reconnaissance de la parole (Speech to text). En effet, la séparation des sources ont des approches similaires dans les deux domaines. De plus, il existe un lien entre partition musicale comme manière d'écrire la musique et texte comme manière d'écrire la parole. La transcription musicale étant la notation d'une œuvre musicale initialement non écrite, l'analogie avec l'écriture de la parole est aisée. Le tableau 1.1 montre des différences et des similitudes entre les deux domaines.

Domaines	Similitudes	Différences
Speech to text AMT	signal $\Rightarrow$ phonèmes $\Rightarrow$ texte signal $\Rightarrow$ notes, accords $\Rightarrow$ partition	données linéaires données structurées

TABLE 1.1 – speechToText vs AMT

Non seulement les objectifs sont similaires, mais les problèmes et les applications, eux aussi, sont comparables (transcription, synthèse, séparation de sources, ...). Il faut néanmoins relever que les informations sont traitées sont de nature différente (voir *mettre ref vers sous-tâches comme beat tracking et inférence de tempo en musique*).

## 1.2 La transcription automatique de la musique

### 1. OBJECTIF

Lorsqu'un musicien est chargé de créer une partition à partir d'un enregistrement et qu'il écrit les notes qui composent le morceau en notation musicale, on dit qu'il a créé une transcription musicale de cet enregistrement. L'objectif de la TAM [11] est de convertir la performance d'un musicien en notation musicale — à l'instar de la conversion de la parole en texte dans le traitement du langage naturel. Cette définition peut être comprise de deux manières différentes selon les articles scientifiques : 1) Processus de conversion d'un enregistrement audio en une notation pianoroll (une représentation bidimensionnelle des notes de musique dans le temps) 2) Processus de conversion d'un enregistrement en notation musicale commune<sup>8</sup> (c'est-à-dire une partition).

### 2. APPLICATIONS

La TAM a des applications multiples [11] dont la plus directe est de donner la possibilité à un musicien de générer la partition d'une improvisation en temps réel afin de pouvoir reproduire sa performance ultérieurement. Une autre application notable est la préservation du patrimoine par exemple dans les styles musicaux où il n'existe peu de partitions (le jazz, la pop, les musiques de tradition orale<sup>9</sup>, ...). La TAM est aussi utile pour la recherche et l'annotation automatique d'informations musicales, pour l'analyse musicologique<sup>10</sup> ou encore pour les systèmes musicaux interactifs.

Un grand nombre de fichiers audio et vidéos musicaux sont disponibles sur le Web, et pour la plupart d'entre eux, il est difficile de trouver les partitions musicales correspondantes, qui sont pourtant nécessaires pour pratiquer la musique, faire des reprises ou effectuer une analyse musicale détaillée.

Mais l'intérêt de la TAM est aussi d'avoir des partitions au contenu exploitable, avec des formats texte ou XML (entre autres...) dont les données sont manipulables, contrairement à de simples images en pdf<sup>11</sup>.

### 3. PROBLÈMES ET MÉTHODES SCIENTIFIQUES

L'analyse de la structure hiérarchique des séquences d'accords par utilisation de modèles grammaticaux s'est avérée très utiles dans les analyses récentes de l'harmonie du jazz [12]. Comme déjà évoqué précédemment, il s'agit d'un problème ancien et difficile. C'est un « graal » de l'informatique

---

8. Ici, on parle de notation occidentale.

9. ethno-musicologie

10. par exemple par la constitution de corpus musicologiques

11. Voir <https://archive.fosdem.org/2017/schedule/event/openscore/> et 0\_slides-Martin.pdf.

musicale. En 1976, H. C. Longuet-Higgins [2] évoquait déjà la représentation musicale en arbre syntaxique dans le but d'écrire automatiquement des partitions à partir de données audio en se basant sur un mimétisme psychologique de l'approche humaine. La tâche de la TAM comprend deux activités distinctes : 1) l'analyse et la représentation d'un morceau de musique ; 2) La génération d'une partition à partir de la représentation du morceau.

### 1.3 La transcription automatique de la batterie

La batterie est née au début du vingtième siècle [13]. C'est donc un instrument récent qui s'est longtemps passé de partition. En effet pour un batteur, la qualité de lecteur lorsqu'elle était nécessaire, résidait essentiellement dans sa capacité à lire les partitions des autres instrumentistes (par exemple, les grilles d'accords et la mélodie du thème en jazz) afin d'improviser un accompagnement approprié que personne ne pouvait écrire pour lui à sa place.

Les partitions de batterie sont arrivées par nécessité avec la pédagogie et l'émergence d'écoles de batterie partout dans le monde. Un autre facteur qui a contribué à l'expansion des partitions de batterie est l'émergence de la musique assistée par ordinateur (MAO). En effet, l'usage de boîtes à rythmes<sup>12</sup> ou de séquenceurs<sup>13</sup> permettant d'expérimenter soi-même l'écriture de rythmes en les écoutant mixés avec d'autres instruments sur des machines a permis aux compositeurs de s'émanciper de la création d'un batteur en lui fournissant une partition contenant les parties exactes qu'ils voulaient entendre sur leur musique.

La batterie a un statut à part dans l'univers de l'AMT puisqu'il s'agit d'instruments sans hauteur (du point de vue harmonique), d'événements sonores auxquels une durée est rarement attribuée et de notations spécifiques (symboles des têtes de notes) [14].

Les applications de la transcription automatique de la batterie (TAB) seraient utiles, non seulement dans tous les domaines musicaux concernés par la batterie dont certains manquent de partitions, notamment les musiques d'improvisation [11], mais aussi de manière plus générale dans le domaine de la RIM : si les ordinateurs étaient capables d'analyser la partie de la batterie dans la musique enregistrée, cela permettrait de faciliter de nombreuses tâches de traitement de la musique liées au rythme. En particulier, la détection et la classification des événements sonores de la batterie par des méthodes informatiques est considérée comme un problème de recherche important et stimulant dans le domaine plus large de la recherche d'informations musicales [14].

---

12. Roland TR-808

13. SQ-1

La TAB est un sujet de recherche crucial pour la compréhension des aspects rythmiques de la musique, et a potentiellement un fort impact sur des domaines plus larges tels que l'éducation musicale et la production musicale.

## 1.4 Les représentations de la musique

### Les partitions

Une partition de musique est un document qui porte la représentation systématique du langage musical sous forme écrite. Cette représentation est appelée transcription.



FIGURE 1.1 – Les figures de notes

La figure 1.1 montre 4 figures de notes les plus courantes dont les noms et les durées, exprimées en unité de temps musicale, appelée le *temps* sont respectivement, de gauche à droite :

- La ronde, elle vaut 4 temps ;
- La blanche, elle vaut 2 temps ;
- La noire, elle vaut 1 temps ;
- La croche, elle vaut 1/2 temps.

Une figure de note [1] de musique combine plusieurs éléments [15]

- Une tête de note :  
Sa position sur la portée indique la hauteur de la note. La tête de note peut aussi indiquer une durée.
- Une hampe :  
c'est une barre verticale liée à la tête de note. Elle est indicatrice d'une durée représentée par sa présence ou non (différence entre la blanche et la ronde)
- Un crochet : La durée d'une note est divisée par deux à chaque crochet ajouté à la hampe d'une figure de note.

La figure 1.2 montre les rapports de durée entre les figures de notes. Plus les durées sont longues, plus elles sont marquées par la tête de note ou la présence ou non de la hampe. À partir de la noire (3ème lignes en partant du haut), on ajoute un crochet à la hampe d'une figure de notes pour diviser sa durée par 2. Les notes à crochet (croche, double-croche, triple-croche...) peuvent être reliées ou non par des ligatures (voir les 4 dernières lignes de la figure 1.2).





FIGURE 1.2 – Rapport des figures de notes  
[1]

	la pause	la demi-pause	le soupir (2)	le demi-soupir	le quart de soupir	le huitième de soupir	le seizième de soupir
Silences							
Notes							
	la ronde	la blanche	la noire	la croche	la double croche	la triple croche	la quadruple croche

FIGURE 1.3 – Les silences

« Les silences sont des signes qui indiquent l'interruption du son. » [1]. La figure 1.3 montre pour chaque figure de note, la figure de silence qui lui correspond.

La transcription sert à traduire les trois caractéristiques du son musical :

- la hauteur, nombre de vibration en un temps donné (son grave ou aigu — do, ré, mi, . . .);
- l'intensité, amplitude des vibrations (force du son);
- le timbre, il permet de différencier deux instruments même s'ils jouent un son de même hauteur et de même intensité.

Ainsi que de leurs combinaisons appelées à former l'ossature de l'œuvre musicale dans son déroulement temporel, à la fois :

- diachronique (succession des instants, ce qui constitue en musique la mélodie);
- et synchronique (simultanéité des sons, c'est-à-dire l'harmonie).

## voix

Pour les instruments mélodiques, un groupe de notes peut être organisé en *voix*, représentant des flots mélodiques joués en parallèle, avec une synchronisation plus ou moins stricte [16] [17].

explications sur l'aspect structuré (hiérarchie) : les mesures, les groupes rythmiques... c'est important ici

### signature rythmique

<flo>présenter rapidement la notation des signatures rythmiques</flo>

### appogiatures

<flo>Parler des appogiatures ici?</flo>

## Les données MIDI

Le format MIDI <sup>14</sup> est originellement une norme technique mais peut également être considéré comme une représentation musicale. Celle-ci peut effectivement être enregistrée par des instruments compatibles, et visualisée ou jouée par un l'ordinateur. Ce format historique, encore très largement utilisé, est très important (mais aussi contraignant) dans le cadre de notre travail, dans la mesure où de nombreux logiciels l'utilisent. Pour la transcription musicale, il constitue une strate intermédiaire très utile entre le signal audio (enregistrement) et la représentation musicale lisible par un humain (partition).

Il s'agit d'un protocole temps réel pour échanger des messages (événement) et un format de fichier. Un fichier MIDI est une séquence d'événements datés et il représente une performance musicale symbolique.

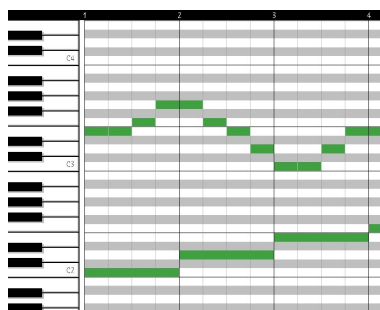


FIGURE 1.4 – Exemple d'événements MIDI

Les données midi sont représentées sous forme de piano-roll. Chaque bande verte sur la figure 1.4 correspond à une notes jouée par un instrument. Le début (onset) et la fin (offset) d'une bande verte sont appelés « événement MIDI ». Il existe des événements ON et des événements OFF qui ont chacun une date sur la séquence MIDI. La durée d'une note est représentée la distance entre les événements ON et OFF qui lui correspondent.

Liste des événements principaux d'un événement MIDI :

14. <https://www.midi.org/>

- type de l'évènement, ON ou OFF ;
- date de l'évènement, Sa position sur la séquence ;
- pitch de l'évènement, à quelle note il correspond (à quelle instrument pour la batterie)
- vitesse, volume sonore de l'évènement.

## Les formats XML

<dam>si tu as la date d'apparition du format, éventuellement une citation un peu originelle</dam> Il existe plusieurs formats XML dédiés à la musique : MusicXML, MEI, MNX, ...

L'inconvénient de ces formats est qu'ils sont verbeux et ambigus, c'est pourquoi nous utilisons pour la transcription une représentation intermédiaire abstraite décrite plus loin.

cela pourra être utile d'avoir une explication (ici ou en 1.4) sur la différence entre les timings de performance (dont le MIDI non-quantifié est un enregistrement symbolique) et les timing des partitions. avec 2 unités temporelles différentes (secondes et temps), en relation par tempo.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE score-partwise PUBLIC
  "-//Recordare//DTD MusicXML 3.1 Partwise//EN"
  "http://www.musicxml.org/dtds/partwise.dtd">
<score-partwise version="3.1">
  <part-list>
    <score-part id="P1">
      <part-name>Music</part-name>
    </score-part>
  </part-list>
  <part id="P1">
    <measure number="1">
      <attributes>
        <divisions>1</divisions>
        <key>
          <fifths>0</fifths>
        </key>
        <time>
          <beats>4</beats>
          <beat-type>4</beat-type>
        </time>
        <clef>
          <sign>G</sign>
          <line>2</line>
        </clef>
      </attributes>
      <note>
        <pitch>
          <step>C</step>
          <octave>4</octave>
        </pitch>
        <duration>4</duration>
        <type>whole</type>
      </note>
    </measure>
  </part>
</score-partwise>
```



FIGURE 1.5 – Exemple de représentation MusicXML

Le figure 1.5<sup>15</sup> représente un do en clef de sol de la durée d'une ronde sur une mesure en 4/4 écrit au format MusicXML. Un des avantages de ce format est qu'il peut être converti aussi bien en données MIDI qu'en partition musicale, ce qui en fait un bon format intermédiaire pour manipuler les données musicales et les échanger entre les programmes. <dam>c'est un peu ce que je mettais plus haut pour le MIDI, là on a une autre représentation, moins "audio" et plus "musique" bref tu peux développer cet aspect de représentation formelle et le questionner un peu. si tu as le temps et l'envie, tu pourrais poser le problème. qu'est-ce qu'une bonne représentation (et informatisable) de la musique ? quels sont les avantages / inconvénients de MIDI / MusicXML ? et enfin, la question centrale que tu pourrais déjà évoquer ici : dans quelle mesure les correspondances MIDI <=> MusicXML sont-elles univoques, ou y-a-t-il plusieurs choix possibles (spoiler oui)</dam>

15. Source images : <https://fr.wikipedia.org/wiki/MusicXML>

## Conclusion

Dans ce chapitre, nous avons établi que la RIM a connu un fort développement ces dernières années, et qu'il s'inspire de plus en plus de méthodes TAL, et que, par ce biais, il y a des liens possibles entre le traitement du langage musical et celui des langues naturelles, le plus proche étant probablement le phénomène de transcription (par analogie avec le *speech to text*).

## ÉTAT DE L'ART

### Sommaire

2.1	Monophonique et polyphonique . . . . .	21
2.2	De l'enregistrement audio vers le MIDI . . . . .	22
2.3	Du format MIDI vers une partition . . . . .	23
2.4	Approche linéaire et approche hiérarchique . . . . .	23

### Introduction

Dans ce chapitre, nous présenterons quelques travaux antérieurs dans le domaine de la transcription automatique de la musique et de la batterie afin de situer notre démarche.

Nous aborderons le passage crucial du monophonique au polyphonique dans la transcription. Nous ferons un point sur les deux grandes parties de la TAM de bout en bout : de l'audio vers le MIDI puis des données MIDI vers l'écriture d'une partition. Ensuite, nous discuterons des approches linéaires et des approches hiérarchiques.

### 2.1 Monophonique et polyphonique

Les premiers travaux en transcription ont été faits sur l'identification des instruments monophoniques<sup>1</sup> [11]. Actuellement, le problème de l'estimation automatique de la hauteur des signaux monophoniques peut être considéré comme résolu, mais dans la plupart des contextes musicaux, les instruments sont polyphoniques<sup>2</sup>. L'estimation des hauteurs multiples est le problème central de la création d'un système de transcription de

1. Instruments produisant une note à la fois, ou plusieurs notes de même durée en cas de monophonie par accord (flûte, clarinette, sax, hautbois, basson, trombone, trompette, cor, etc...)

2. guitare, piano, basse, violon, alto, violoncelle, contrebasse, glockenspiel, marimba, etc...

musique polyphonique. Tout signal audio musical peut être composé de plusieurs signaux, ceux-ci pouvant provenir de plusieurs instruments, ou d'un instrument dit polyphonique. Une tâche difficile consiste à séparer, à partir du signal, les différentes sources (ou voix) afin de les représenter individuellement. La batterie, composée de plusieurs instruments (caisse claire, grosse caisse, cymbales, toms, etc...), est un cas typique d'instrument polyphonique pour lequel ce défi est majeur.

Les performances des systèmes actuels ne sont pas encore suffisantes pour permettre la création d'un système automatisé capable de transcrire de la musique polyphonique sans restrictions sur le degré de polyphonie ou le type d'instrument. Cette question reste donc encore ouverte.

## 2.2 De l'enregistrement audio vers le MIDI

Jusqu'à aujourd'hui, les recherches se sont majoritairement concentrées sur le traitement de signaux audio vers la génération de contenu MIDI non-quantifié (une performance — à expliquer) [18]. Cette tâche englobe plusieurs sous-tâches dont séparation des sources audio, la détection multi-pitches<sup>3</sup> et la détection des *onsets* et des *offsets*.

En transcription automatique de la batterie [14], plusieurs stratégies de répartition pré/post-*processing* sont possibles pour la détection multi-pitches. La détection peut être entamée dès le pré-*processing*, en supprimant les *features*<sup>4</sup> non-pertinentes pendant la séparation des sources afin d'obtenir une meilleure détection des instruments de la batterie, par exemple en supprimant la structure harmonique pour atténuer l'influence des instruments à hauteurs sur la détection grosse caisse et caisse claire. Mais certaines études montrent que la suppression des instruments à hauteurs peut avoir des effets néfastes sur les performances de la transcription de batterie. En outre, les systèmes de TAB basés sur des réseaux de neurones récurrents ou sur des factorisations matricielles font la séparation des sources pendant l'optimisation, ce qui réduit la nécessité de la faire pendant le pré-*processing*. Pour la reconnaissance des instruments de la batterie, une autre approche possible est d'utiliser un modèle probabiliste (MMC) pour classifier les différents sons de la batterie [19]. L'approche AdaMa [20], qui commence par une estimation initiale des sons de la batterie en les raffinant itérativement, est une autre approche de la même catégorie.

avant la TAB, il faudrait dire 2 mots sur les techniques utilisées (cf. survey AMT Benetos et al.)

3. La détection multi-pitches est la détection des hauteurs simultanées pour les instruments polyphoniques. Il peut s'agir de notes d'un même instrument ou de plusieurs instruments différents.

4. Features : caractéristiques individuelles mesurables d'un phénomène dans le domaine de l'apprentissage automatique et de la reconnaissance des formes

Toutes ces méthodes elles visent toutes la génération d'un contenu MIDI non-quantifié qui est la représentation symbolique d'une performance musicale.

## 2.3 Du format MIDI vers une partition

Les approches mentionnées en section 2.2 produisent en sortie un fichier MIDI non-quantifié, qui est un format encore très éloigné d'une partition musicale. Un premier problème concerne les timings (dates et durées d'événements) qui doivent être alignées à des positions temporelles correspondant à des valeurs exprimables avec la notation musicale (voir la différence entre contenu MIDI et musique écrite en section 1.4). On parle de quantification rythmique.

Nakamura et al. 2016 présentent une approche de quantification rythmique avec modèles de probabilités (MMC) qui prend en entrée un fichier MIDI non quantifié et fourni en sortie un fichier MIDI quantifié. [16] étendent ensuite l'approche à une transcription d'enregistrement audio vers un fichier MIDI quantifié. Ce dernier format, linéaire, ne correspond toutefois pas encore à une partition structurée, avec groupement rythmiques hiérarchiques (voir la section 1.4). Dans ces travaux, la structuration des données en partition est déléguée à un éditeur de partitions (MuseScore), avec des résultats assez inégaux.

Seuls quelques travaux récents s'intéressent de près à la création d'outils permettant la génération de partition. Le problème de la conversion d'une séquence d'événements musicaux symboliques en une partition musicale structurée est traité notamment dans [21]. Ce travail, qui vise à résoudre de manière conjointe la quantification rythmique et la production de partitions structurées, s'appuie tout au long du processus sur des grammaires génératives qui fournissent un modèle hiérarchique — langage a priori des partitions. Les expériences ont des résultats prometteurs, mais il faut relever qu'elle ont été menées avec un ensemble de données composé d'extraits monophoniques ; Il reste donc à traiter le passage au polyphonique, qui nécessite de traiter le problème supplémentaire de la séparation de voix, en le couplant avec la quantification du rythme.

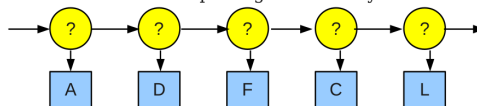
L'approche de [21] est fondée sur la conviction que la complexité de la structure musicale dépasse les modèles linéaires.

i.e. pour la batterie on a besoin de quantification + structuration + séparation mais seules les 2 premières sont couplées dans l'approche de ton stage.

## 2.4 Approche linéaire et approche hiérarchique

La figure 2.1 montre un exemple de modèle linéaire (le modèle de Markov caché). Ce type de modèle traite les événements localement les uns après les autres. La figure mentionne entre autres que les MMC sont utilisés en reconnaissance de la parole. Il est intéressant de relever que la repré-

- Modèle de Markov **caché** :
  - **Hidden Markov Model (HMM) (Baum, 1965)**
  - Modélisation d'un processus stochastique « **génératif** » :
    - État du système : non connu
    - Connaissance pour chaque état des **probabilités** comme état initial, de **transition** entre états et de **génération** de symboles
    - **Observations** sur ce qu'a « généré » le système

FIGURE 2.1 – Le modèle de Markov caché<sup>5</sup>

sentation écrite de la parole est linéaire puisque les évènements sont lus les uns après les autres avant que l'on puisse percevoir la structure d'une idée. Ce dernier point constitue une différence notable entre l'écriture de la parole et notation musicale est puisque cette dernière est explicitement structurée.

Plusieurs travaux ont d'abord privilégié l'approche stochastique. Par exemple, Shibata *et al.* [16] ont utilisé des MMC pour la reconnaissance des signatures rythmiques. Les auteurs utilisent d'abord deux réseaux de neurones profonds, l'un pour la reconnaissance des *pitchs* et l'autre pour la reconnaissance de la vélocité. Ils construisent ensuite plusieurs MMC étendus pour la musique polyphonique correspondant à des signatures rythmiques potentielles afin de trouver la signature la plus probable. L'évaluation finale des résultats de [16] montre qu'il faut rediriger l'attention vers les valeurs des notes, la séparation des voix et d'autres éléments délicats de la partition musicale qui sont significatifs pour son interprétation.

Même si la quantification du rythme se fait le plus souvent localement en manipulant des données linéaires, par déduction de la probabilité d'une durée à partir de la durée précédente (par exemple dans un MMC), de nombreux travaux suggèrent d'aborder le problème plus globalement en utilisant une approche hiérarchique puisque le langage musical est lui-même structuré.

En effet, l'utilisation d'arbres syntaxiques semble appropriée pour représenter le langage musical. Une méthodologie simple pour la description et l'affichage des structures musicales est présentée dans [22]. Les arbres de rythmes y sont évoqués comme permettant une cohésion complète de la notation musicale traditionnelle avec des notations plus complexes. Jacquemard *et al.* [23] propose aussi une représentation formelle du rythme,

5. Source : cours de Damien Nouvel <https://damien.nouvel.net/fr/enseignement>



inspirée de modèles théoriques antérieurs issus du domaine des techniques de réécriture appliquées à la déduction automatique et au calcul symbolique. Ils montrent aussi qu'il est possible d'appliquer des arbres de rythmes pour le calcul d'équivalences rythmiques dans [24]. La réécriture d'arbres, dans un contexte de composition assistée par ordinateur, par exemple, pourrait permettre de suggérer à un utilisateur diverses notations possibles pour une valeur rythmique, avec des complexités différentes.

La nécessité d'une approche hiérarchique pour la production automatique de partition est évoquée dans [21]. Les modèles de grammaire qui y sont exposés sont différents des modèles markoviens linéaires de précédents travaux.

citer thèse de David Rizo  
(Valencia)

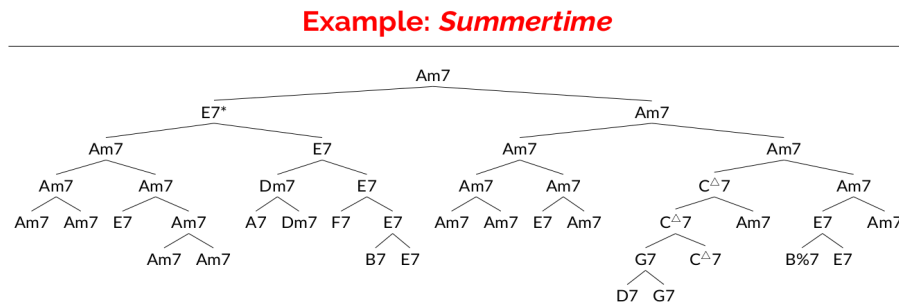


FIGURE 2.2 – Représentation arborescente d'une grille harmonique [12]

La figure 2.2 est une représentation dans un arbre syntaxique de la structure harmonique du standard de jazz summertime. La racine de l'arbre est la tonalité du morceau. Entre la racine et les feuilles se trouve la structure harmonique qui défile en fonction des différentes parties du morceau. Les feuilles sont jouées les accords joués.

## Conclusion

La plupart des travaux déjà existants sur la TAB ont été énumérés par Wu *et al.* [14] qui, pour mieux comprendre la pratique des systèmes de TAB, se concentrent sur les méthodes basées sur la factorisation matricielle et celles utilisant des réseaux neuronaux récurrents. La majorité de ces recherches se concentre sur des méthodes de calcul pour la détection d'événements sonores de batterie à partir de signaux acoustiques ou sur la séparation entre les événements sonores de batterie avec ceux des autres instruments dans un orchestre ou un groupe de musique [25], ainsi que sur l'extraction de caractéristiques de bas niveau telles que la classe d'instrument et le moment de l'apparition du son. Très peu d'entre eux ont abordé la tâche de générer des partitions de batterie et, même

quand le sujet est abordé, l'output final n'est souvent qu'un fichier MIDI non-quantifié et non une partition écrite.

En conclusion, il n'existe pas de formalisation de la notation de la batterie ni de réelle génération de partition finale, dont les enjeux principaux seraient :

1. le passage du monophonique au polyphonique, comprenant la distinction entre les sons simultanés et les appogiatures ou autres ornements ;
2. les choix d'écritures spécifiques à la batterie concernant la séparation des voix et les continuations.

## MÉTHODES

### Sommaire

3.1	La notation de la batterie . . . . .	27
3.2	La transcription manuelle . . . . .	35
3.3	Modélisation pour la transcription . . . . .	37
3.4	Analyse syntaxique pour la transcription . . . . .	39
3.5	Les <b>formes rythmiques</b> . . . . .	42

### Introduction

Dans ce chapitre, nous expliquerons en détail les méthodes que nous avons employées pour l'ADT.

Pour commencer, nous exposerons une description de la notation de la batterie ainsi qu'une modélisation de celle-ci pour la représentation des données rythmiques en arbres syntaxiques. Nous poursuivrons avec une présentation de `qparse`<sup>1</sup>, un outil de transcription qui est développé à l'Inria, l'Université de Nagoya et plusieurs développeurs au sein du laboratoire Cedric au CNAM.

Enfin, nous présenterons les **formes rythmiques**, <flo>une représentation théorique qui permet. . .</flo>

### 3.1 La notation de la batterie

Pour la transcription, j'ai choisi d'utiliser une notation inspirée du recueil de pièces pour batterie de J.-F. Juskowiak [26] et des méthodes de batterie Agostini [27], car je trouve la position des éléments cohérente et intuitive (voir section ??).

---

1. <https://qparse.gitlabpages.inria.fr/>

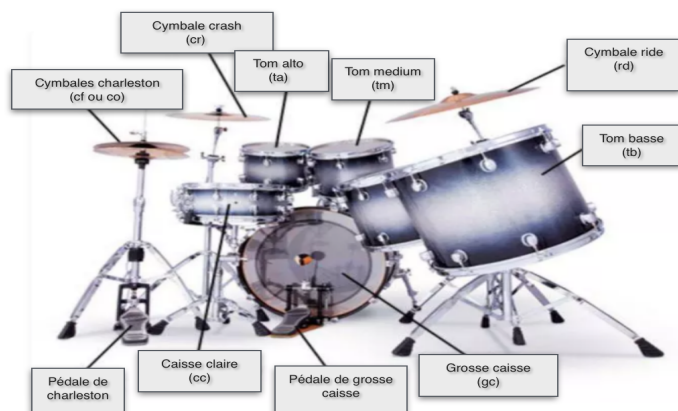


FIGURE 3.1 – Les instruments de la batterie

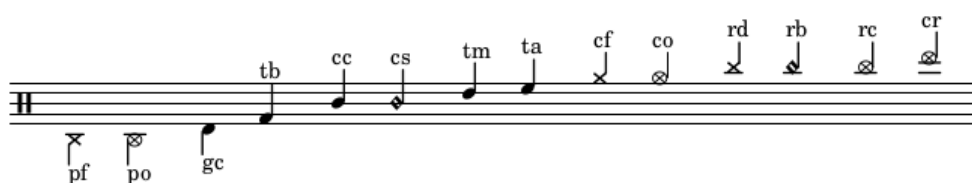


FIGURE 3.2 – Hauteur et têtes de notes

Noms figure 3.1	codes figure 3.2	référence
Pédale de charleston	pf ou po	charley fermé ou ouvert au pied
Grosse caisse	gc	grosse caisse
Tom basse	tb	tom basse
Caisse claire	cc	caisse claire
Tom médium	tm	tom médium
Tom alto	ta	tom alto
Cymbales charleston	cf ou co	charley fermé ou ouvert à la main
Cymbales ride	rd	ride
Cymbales crash	cr	crash

TABLE 3.1 – Noms des instruments de la batterie

La figure 3.1<sup>2</sup> montre une batterie standard avec tous les instruments habituellement présent sur une batterie et la figure 3.2 donne leur représentation sur une partition.

Le tableau 3.1 donne dans l'ordre :

1. les noms des instruments sur la figure 3.1 ;
2. leurs codes respectifs dans la figure 3.2 ;
3. les noms que j'utiliserai dans le présent document pour y référer.

Les figures 3.1, 3.2 et le tableau 3.1 peuvent aider à comprendre pourquoi je trouve la notation agostinienne cohérente et intuitive.

En effet, les hauteurs sur la portée représentent :

1. La hauteur physique des instruments :  
La caisse claire est centrale sur la portée et sur la batterie (au niveau de la ceinture, elle conditionne l'écart entre les pédales et aussi la position de tous les instruments basiques d'une batterie).  
Tout ce qui en-dessous de la caisse claire sur la portée est en dessous de la caisse claire sur la batterie (pédales, tom basse) ;  
Tout ce qui est au-dessus de la caisse claire sur la portée, l'est aussi sur la batterie.
2. La hauteur des instruments en terme de fréquences :  
Sauf pour le charley au pied et si l'on sépare en trois groupes (grosse caisse, toms et cymbales), de bas en haut, les instruments vont du plus grave au plus aigu.

## Les durées

Comme nous venons de la voir, la majorité des instruments de la batterie sont représentés par les têtes des notes. De plus, le seul instrument dont le son peut être arrêté de manière quantifiée et dont la durée sonore nous intéresse est le charley<sup>3</sup>.

Par conséquent :

1. les durées — sauf pour le charley — représenteront un écart temporel entre les notes et non une durée sonore et elles pourront donc être rallongée à l'aide de silences ;
2. les symboles rythmiques concernant les têtes de note ne pourront pas être utilisés pour exprimer les durées. Cela est valable aussi pour la présence ou non de la hampe puisque ce phénomène n'existe qu'avec les têtes de notes de type cercle-vide (opposition blanche-ronde). L'usage des blanches existe dans certaines partitions de

certaines têtes de notes vides alors que leur durée n'est pas celle des blanches ? expliquer les différences avec la notation conventionnelle cf 1.4

2. Source : <https://www.superprof.fr/blog/composition-instrument-percussion/>

3. Je ne prendrais pas en compte l'arrêt des cymbales à la main car ce phénomène n'existe pas dans les fichiers MIDI.

batterie [28] mais cela reste dans des cas très rares. Certains logiciels permettent de faire des blanches avec des symboles spécifiques à la batterie ou aux percussions mais leur lecture reste peu aisée et leur utilisation pour la batterie est rarissime.

En résumé :

- toutes les notes ont une hampe ;
- une notes dont la hampe n'a pas de crochet est toujours une noire ;
- à part pour le charley ouvert, les durées n'expriment pas la durée d'un son mais une distance temporelle entre deux notes.
- à part pour le charley ouvert, la durée d'une note peut être prolongée par un silence (exemple : une noire + un soupir pour exprimer une blanche)

La durée d'une note peut être prolongée par divers symboles :

- Le point : il rallonge la durée d'une note de la moitié de sa valeur. Dans la deuxième note de l'exemple 3 de la figure 3.3 est une noire pointée, elle vaut donc la durée d'une noire + une croche (ou de trois croches) ;
- La liaison : elle rallonge la durée de la première note de la durée de la deuxième. La deuxième note de l'exemple 4 de la figure 3.3 est une croche qui est liée à une noire, sa durée est donc équivalente à celle d'une croche + une noire (ou de trois croches) ;
- les silences (pas pour les ouvertures de charley).



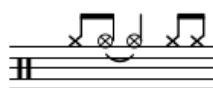
Exemple 1



Exemple 2



Exemple 3



Exemple 4

FIGURE 3.3 – Point et liaison

Un autre élément concernant la notation des durées en batterie est la nécessité de faire ressortir la pulsation<sup>4</sup> de la rendre visuelle. La première chose à prendre en compte pour analyser la figure 3.3 est donc la nécessité de regrouper les notes par temps à l'aide des ligatures. Le deuxième point est de s'arranger pour qu'il y ait une indication visuelle au début de chaque temps.

- Exemple 1 : l'ouverture de charley est quantifiée mais les notes ne sont pas regroupées par temps.
- Exemple 2 : Ici, la liaison permet de regrouper les notes par temps en obtenant le même rythme que dans l'exemple 1.
- Exemple 3 et exemple 4 : les deux exemples sont valables mais le deuxième est le plus souvent utilisé car la liaison donne un repaire visuel sur le temps.

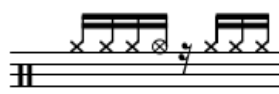
En cas de nécessité de prolonger la durée d'une note au-delà de son temps de départ (syncope) et si cette note ne correspond pas à une ouverture de charley, elle sera prolongée sur le temps suivant à l'aide de silences dont le premier sera positionné sur le temps. Si la note syncopée est une ouverture de charley, on privilégiera la liaison pour sa prolongation.

## Les silences

Les silences sont parfois utilisés pour noter les fermetures de charley (après une ouverture). Les fermetures du charley sont notées soit par un silence (correspondant à une fermeture de la pédale), soit par un écrasement de l'ouverture par un autre coup de charley fermé, au pied ou à la main.



Exemple 1



Exemple 2

FIGURE 3.4 – Silence joué

L'écriture littérale de contenu MIDI peut ressembler à l'exemple 1 de la figure 3.4. Sur cet exemple, le son de l'ouverture de charley est arrêté par une pression du pied sur la pédale et c'est ce que le batteur joue dans les faits. Mais il apparaît intuitivement que le but de la première note

4. La position des temps

du deuxième temps n'est pas de générer un son de charley au pied mais uniquement de stopper l'ouverture. La notation de l'exemple 2 de la figure 3.4 serait donc préférable car elle représente mieux l'intention de ce rythme et elle n'empiète pas sur une potentielle voix basse qui pourrait le compléter (on évite une écriture surchargée).

Lorsqu'une note est un charley ouvert, il faudra donc prendre en compte la note suivante pour l'écriture :

1. si c'est un charley fermé joué à la main  $\Rightarrow$  la note sera un charley fermé joué à la main (cf) ;
2. si c'est un charley fermé joué au pied  $\Rightarrow$  la note sera un silence.

La deuxième règle sera soumise au cadre imposé par certaines **formes rythmiques** pour lesquelles le charley joué au pied devra rester tel quel.

### Les équivalences rythmiques

Pour les instruments mélodiques, dans le cas de notes dont la durée de l'une à l'autre est ininterrompue et si leur durée initiale est prolongée, seuls la liaison et le point permettent des notations équivalente. Mais pour la batterie et à part dans le cas des ouvertures de charley (voir section 3.1), seules comptent des dates de début (onsets) : la durée du son n'a pas d'importance. L'usage des silences pour combler la distance rythmique entre deux notes devient donc possible.

Cela pris en compte, et étant donné que les indications de durée dans les têtes de notes sont peu recommandées (voir section 3.1), l'écriture à l'aide de silences sera privilégiée comme indication de durée sauf dans les cas où cela reste impossible. Ce choix a pour but de n'avoir qu'une manière d'écrire toutes les notes, quelles que soient leur tête de note (sauf pour le charley).



FIGURE 3.5 – Équivalence

Sur la figure 3.5, théoriquement, il faudra choisir la notation de la deuxième mesure mais dans certains contextes, pour des raisons de lisibilité ou de surcharge, la version sans les silences de la troisième mesure pourra être choisie.



## Les voix

En batterie, une voix est théoriquement l'ensemble des instruments qui, à eux seuls, constituent une phrase rythmique. Mais en pratique, les instruments peuvent aussi être divisés par voix dans le but de ne pas surcharger la notation ou pour que leur disposition soit représentée sur la partition (voir section 3.1). Les voix sont caractérisées par l'orientation des hampes et plus précisément par les ligatures si les hampes sont dans la même direction (voir figure 3.18).



FIGURE 3.6 – Séparation des voix

Sur la figure 3.6, il faudra faire un choix entre les exemples 1, 2 et 3 qui sont trois façons équivalentes d'écrire le même rythme. Ce choix se fera en fonction des instruments joués, de la nature plus ou moins systématique de leurs phrasés, et des associations logiques entre les instruments dans la distribution des rythmes sur la batterie (voir la section 3.5).

## Les accentuations et les ghost-notes

« Certaines notes dans une phrase musicale doivent, ainsi que les différentes syllabes d'un mot, être accentuées avec plus ou moins de force, porter une inflexion particulière. » [1]

Théoriquement, tous les instruments peuvent être accentués (voir la section 3.3), mais la figure 3.7 représentent ceux dont les accents ne demandent pas un grand niveau de maîtrise et sont presque toujours bien articulés. En outre, les instruments qui ne sont pas représentés sur cette figure ne sont presque jamais accentués dans les partitions et ne sont pas présents de manière significative dans le GMD.

Les accents sont marqués par le symbole « > ». Ils sont positionnés au-dessus des notes représentant des cymbales et en-dessous des notes re-



FIGURE 3.7 – Les accents et les ghost-notes

présentant des toms ou la caisse claire. Ce choix a été fait pour la partition de la figure 3.11 car elle est plus lisible ainsi, mais ces choix devront être adaptés en fonction des différentes **formes rythmiques** reconnues (voir la section 3.5). Par exemple, pour les **formes rythmiques** jazz, les ligatures pour les toms et la caisse claire seront dirigées vers le bas, il faudra donc mettre les symboles d'accentuation correspondants au-dessus des têtes de notes.

La dernière note de la figure 3.7 montre un exemple de notation pour une ghost note jouée à la caisse claire. Une ghost note [29] est une note de faible volume sonore mais jouée fermement. Les ghost notes servent le plus souvent à donner le débit d'un rythme (ses subdivisions) pour le rendre plus dansant (lui donner plus de « groove » ou de « swing »). Le parenthésage a été choisi car il peut être utilisé sur n'importe quelle note sans changer la tête de note.

Toutes les notes de la figure 3.7 sont exposées en situation réelle dans la figure 3.8.



FIGURE 3.8 – Exemple pour les accentuations et les ghost-notes

### Les flas

Le fla est appoggiature qui consiste à jouer deux coups presque simultanés dont le premier est une ghost note et le deuxième une note normale ou accentuée.



FIGURE 3.9 – Définition du fla

## 3.2 La transcription manuelle

Mis à part les figures du chapitre 1 et certains exemples d'analyses de la section 4.2, toutes les partitions et figures de ce document ont généré avec lilypond<sup>5</sup>.

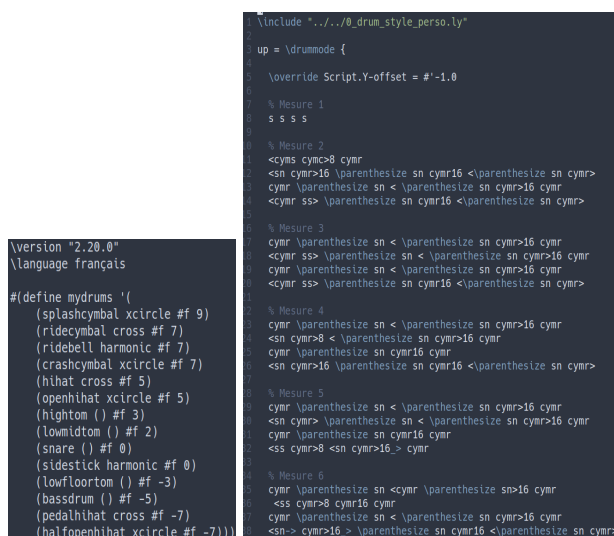
### Présentation de lilypond

« LilyPond est un logiciel de gravure musicale, destiné à produire des partitions de qualité optimale. Ce projet apporte à l'édition musicale informatisée l'esthétique typographique de la gravure traditionnelle. LilyPond est un logiciel libre rattaché au projet GNU. »

En raison de :

- notation agostini
- grande liberté de choix
- ...

Lilypond est actuellement le meilleur de logiciel de gravure musicale pour la batterie.



```

\version "2.20.0"
\language français

#(define mydrums '(
  (splashcymbal xcircle #f 9)
  (ridecymbal cross #f 7)
  (ridebell harmonic #f 7)
  (crashcymbal xcircle #f 7)
  (hihat cross #f 5)
  (openhihat xcircle #f 5)
  (hightom () #f 3)
  (lowmidtom () #f 2)
  (snare () #f 0)
  (sidestick harmonic #f 0)
  (lowfloortom () #f -3)
  (bassdrum () #f -5)
  (pedalhihat cross #f -7)
  (halfopenhihat xcircle #f -7)))

\include "../.../0_drum_style_perso.ly"

up = \drummode {

\override Script.Y-offset = #-1.0

% Mesure 1
s s s

% Mesure 2
<cymr> cymc>8 cymr
<sn cymr>16 \parenthesize sn cymr16 <\parenthesize sn cymr>
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<cymr> ss> \parenthesize sn cymr16 <\parenthesize sn cymr>

% Mesure 3
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<cymr> ss> \parenthesize sn <\parenthesize sn cymr>16 cymr
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<cymr> ss> \parenthesize sn cymr16 <\parenthesize sn cymr>

% Mesure 4
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<sn cymr>8 <\parenthesize sn cymr>16 cymr
cymr \parenthesize sn cymr16 cymr
<sn cymr>16 \parenthesize sn cymr16 <\parenthesize sn cymr>

% Mesure 5
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<sn cymr> \parenthesize sn <\parenthesize sn cymr>16 cymr
cymr \parenthesize sn cymr16 cymr
<ss cymr>8 <sn cymr>16.> cymr

% Mesure 6
cymr \parenthesize sn <cymr> \parenthesize sn>16 cymr
<ss cymr>8 cymr16 cymr
cymr \parenthesize sn <\parenthesize sn cymr>16 cymr
<sn> cymr>16.> \parenthesize sn cymr16 <\parenthesize sn cymr>

```

FIGURE 3.10 – lilypond — extraits de code

Sur la figure 3.10 :

- à gauche : configuration aménagée pour la notation de type agostini.
- à droite : le début de code mesure par mesure pour la voix haute d'une partition (en haut du fichier, inclusion du fichier de config)

5. <http://lilypond.org/index.fr.html>

FIGURE 3.11 – lilypond — transcription manuelle

La partition de la figure 3.11 est le résultat du code de la figure 3.10 (la totalité du code est mis en annexe et est accessible sur le git). Cette partition a été totalement transcrite manuellement avec lilypond par analyse des fichiers MIDI et audio correspondants.

- difficultés principales : trouver une application permettant de choisir librement la notation de la batterie. Lilypond le permet mais beaucoup de recherches ont été nécessaires pour comprendre l'ensemble des fonctionnalités permettant de faire fonctionner la notation « agostinienne » ainsi que les diverses subtilités de notations (accents, ghost-notes, flas, ...).
- lilypond reste néanmoins un choix très agréable, une fois ces difficultés surmontées.
- Écrire la partition de la figure 3.11 m'a pris beaucoup de temps car j'ai dû chercher comment écrire chaque nouvel événement mais les autres transcriptions ont été beaucoup plus rapide et très aisées.

- Même si cela représente un investissement au départ, je recommande lylipond pour écrire la batterie et je pense que c'est meilleur outil pour cette tâche pour le moment. On peut configurer absolument tout.
- dans les autres logiciel d'édition de type musescore, la batterie est toujours confiné au système de notation américain.
- pour une comparaison entre système américain et système agostinien, voir section 4.2 est comparer les notations TM (agostinien) et TA (américain).

### 3.3 Modélisation pour la transcription

#### Les pitches

Codes	Instruments	Pitches
cf	charley-main-fermé	22, 42
co	charley-main-ouvert	26
pf	charley-pied-fermé	44
rd	ride	51
rb	ride-cloche (bell)	53
rc	ride-crash	59
cr	crash	55
cc	caisse claire	38, 40
cs	cross-stick	37
ta	tom-alto	48, 50
tm	tom-medium	45, 47
tb	tom-basse	43, 58
gc	grosse caisse	36

TABLE 3.2 – Codes, identités et pitches des instruments

Le tableau 3.2 présente dans l'ordre, les codes des instruments, leur identité (instrument ou parti d'un instrument — joué avec les mains ou avec les pieds), le ou les pitches qui lui sont associés.

Plusieurs pitches peuvent parfois désigner le même instrument afin de pouvoir supporter des kits de batterie plus larges (avec par exemple plusieurs toms basses qui n'auraient pas tous exactement la même sonorité) ou simplement de styles différents (pour chaque kits standard, ce sont les mêmes instruments mais de styles différents)<sup>6</sup>. J'ai regroupé les pitches des différents types d'un même instrument dans une seule ligne du tableau portant le nom du type de cet instrument. Ainsi, plusieurs toms basses différents dans les données MIDI deviennent tous un tom basse d'une

6. Par exemple, les peaux des toms jazz raisonnent alors que les toms rock sont mat.

batterie standard et la partition finale pourra être jouée sur n'importe quel kit de batterie standard.

Malgré le large panel de pitches disponibles, il semblerait qu'aucun pitch ne désigne le charley ouvert joué au pied (« po » de la figure 3.2). Pourtant, dans la batterie moderne, plusieurs rythmes ne peuvent fournir le son du charley ouvert qu'avec le pied car les mains jouent autre chose en même temps. Cela doit en partie être dû à l'utilisation des boîtes à rythmes en MAO qui ne nécessitent pas de faire des choix conditionnés par les limitations humaines (2 pieds, 2 mains, et beaucoup plus d'instruments...)

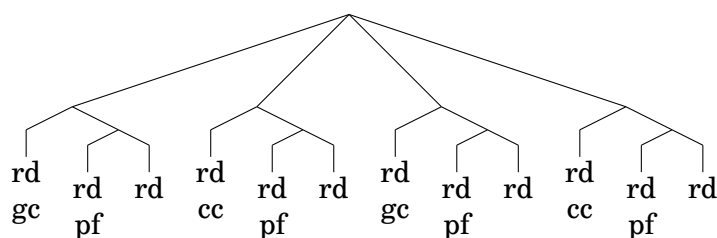
### La vitesse

La vitesse déterminera si les notes sont accentuées ou sont des ghost notes. Pour les codes, je propose d'ajouter un suffix (« a » pour accent et « g » pour ghost note) à la fin du code d'une note accentuée ou d'une ghost note. Les choix pour déterminer si les notes sont accentuées ou sont des ghost notes seront donnée dans la section 4.2.

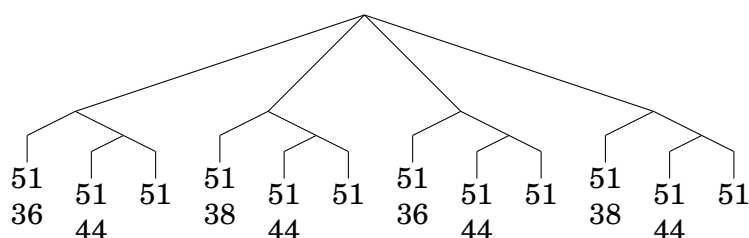
### Les arbres de rythmes

Les arbres de rythmes représentent un rythme dont les possibilités de notation sur une partition sont théoriquement multiples. Les branchements sont des divisions d'intervalle temporel, les feuilles sont des événements musicaux commençant au début de l'intervalle [30] [31] .

Voici une représentation qui fonctionne avec les 3 exemples de la figure 3.6 en arbre de rythmes avec les codes de chaque instrument :



Ci-dessous, le même arbre dont les codes des instruments sont remplacés par leurs données MIDI respectives :



Chacun des trois exemples de la figure 3.6 est représenté par un des deux arbres syntaxiques ci-dessus. <dam>complète un peu en précisant qu'on voit bien ici l'avantage des arbres pour analyser ou construire la structure (les phrases?) musicale</dam>

### 3.4 Analyse syntaxique pour la transcription

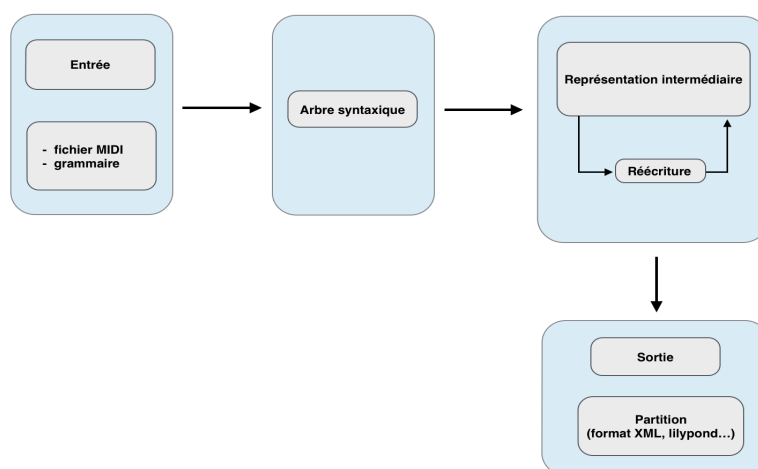


FIGURE 3.12 – Présentation de Qparse

Comme le montre la figure 3.12, *qparse*<sup>7</sup> est un outil pour la transcription musicale, qui, à partir d'une performance symbolique, séquentielle et non quantifiée, produit une partition structurée. Il effectue conjointement des tâches de quantification rythmique et d'inférence de la structure de la partition à l'aide de techniques d'analyse syntaxique (*parsing*). Le but du *parsing* est en effet la structuration d'une représentation séquentielle en entrée (un mot fini), suivant un modèle de langage [32].

Dans le cas de *qparse*, le "mot" d'entrée est typiquement au format MIDI, et le modèle de langage est une grammaire d'arbres pondérés représentant des préférences en terme de notation musicale à produire [33]. basée sur des algorithmes d'analyse syntaxique pour les grammaires arborescentes pondérées. En prenant en entrée une performance musicale symbolique (séquence de notes avec dates et durées en temps réel, typiquement un fichier MIDI), et une grammaire hors-contexte pondérée décrivant un langage de rythmes préférés, il produit une partition musicale. Plusieurs formats de sortie sont possibles, dont les formats XML (MEI, MusicXML,...), lilypond,...

7. <https://qparse.gitlabpages.inria.fr>

Les principaux contributeurs sont :

- Florent Jacquemard (Inria) : développeur principal.
- Francesco Foscari (PhD, CNAM) : apprentissage ; Evaluation.
- Clement Poncelet (Salzburg U.) : integration de la librairie Midifile pour les input MIDI.
- Philippe Rigaux (CNAM) : production de partition au format MEI et de modèle intermédiaire de partition en sortie.
- Masahiko Sakai (Nagoya U.) : mesure de la distance input/output pour la quantification et CMake framework ; évaluation.

### Les enjeux

Plusieurs enjeux : <dam>idem à rédiger je suppose... un assez gros morceau puisqu'il contiendra une partie de ta problématique appliquée</dam> Le problème du MIDI avec Qparse  
ON-OFF en entrée  $\Rightarrow$  1 seul symbole en sortie.  
Minimiser la distance entre le midi et la représentation en arbre.  
Un des problèmes de Qparse était qu'il était limité au monophonique.  
Quelles sont les limites du monophonique ?  
Impossibilité de traiter plusieurs voix et de reconnaître les accords.

### La grammaire

Il s'agit d'une grammaire hors-contexte pondérée dont la structure agit comme un langage a priori de notation rythmique. Les règles de la grammaire et certaines valeurs de poids indiquent des rythmes et des écritures (pour les ligatures notamment) à favoriser.

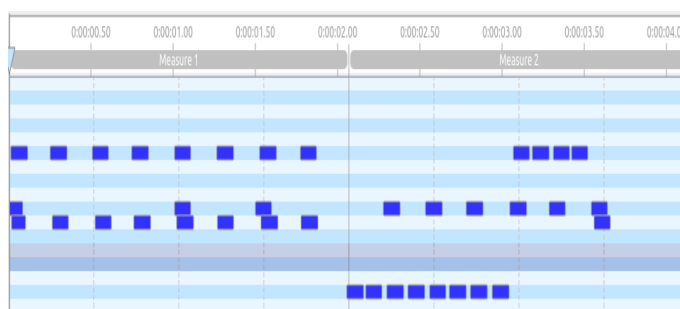


FIGURE 3.13 – Fichier MIDI pour les tests de grammaires

Les exemples suivants ont été écrits pour le fichier MIDI de la figure 3.13. Ce fichier contient les informations de trois mesures dont les deux premières contiennent des noires, des croches (et des doubles-croches pour la deuxième) et dont la troisième est vide. La signature rythmique est 4/4.



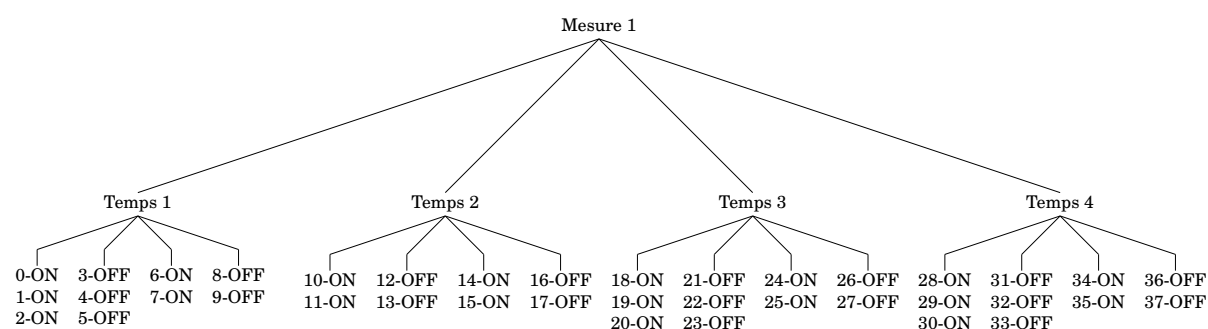
0 -> C0	1
0 -> E1	1
0 -> U4 (1, 1, 1, 1)	1
1 -> C0	1
1 -> E1	1
1 -> T2 (2, 2)	1
1 -> T4 (4, 4, 4, 4)	1
2 -> C0	1
2 -> E1	1
4 -> C0	1
4 -> E1	1
4 -> E2	1

FIGURE 3.14 – Exemple de grammaire

Sur la figure 3.14, chaque ligne est une règle. La colonne de gauche contient des symboles non-terminaux (0, 1, 2, 4).

Cette grammaire sépare les temps par ligatures au niveau de la mesure. Puis elle autorise, au niveau du temps, les divisions par deux (croches) et par quatre (doubles-croches). Ces divisions seront reliés par des ligatures. Tous les poids sont réglés sur 1 à titre expérimental. L'arbre de parsing en résultant est considéré comme « convaincant » car il découpe correctement les mesures et les temps.

Un premier test convaincant est effectué avec la grammaire suivante :



## Le parsing

Les données MIDI sont quantifiées, les notes de dates proches sont alignées et les relations entre les notes sont identifiées (accords, fla, etc...); un arbre syntaxique global est créé ;

### La représentation intermédiaire

- Les pitches sont remplacés par les codes des instruments (voir tableau 3.1 ;
  - réécriture 1 :  
séparation des voix  $\Rightarrow$  un arbre par voix  $\Rightarrow$  représentation intermédiaire (RI) ;
  - réécriture 2 :  
simplification de l'écriture de chaque voix dans la RI.
- <dam>rédigier un peu mieux réécriture 1 et 2</dam>

## 3.5 Les formes rythmiques

Il existe en batterie des motifs rythmiques répétés (joués en boucle). Ces motifs sont le résultat de la coordination de plusieurs instruments de la batterie, je les nommerai **motifs** dans la suite du document. Très souvent, un autre instrument est joué de manière indépendante sur le **motif** mais en respectant la cohérence rythmique du **motif**. Je nommerai **gamme** l'ensemble des combinaisons possibles pour cet autre instrument. La **gamme** d'un **motif** est toujours relative à sa signature rythmique. Enfin, j'ai nommé **forme rythmique (FR)** le couple **motif-gamme**.

### Objectifs

Le but est d'avoir des schémas types (les **formes rythmiques**) pour calculer la séparation en voix. Cela constituerait une heuristique pour éviter d'avoir à explorer une grande combinatoire.

Un ensemble de **formes rythmiques** comprenant leur signatures rythmiques respectives et leurs règles spécifiques de réécriture sera nécessaire.

Une fois un système défini, la transcription à partir de l'entrée MIDI sera facilitée par la reconnaissance de la FR qui contraindra le processus, il permettra de :

- définir une signature rythmique ;
- choisir une grammaire appropriée ;
- fournir les règles de réécriture (séparation des voix et simplification).

### Définitions

- **motif** : rythmes coordonnés joués avec deux ou trois instruments coordonnés en boucle (répartis sur 1 ou 2 voix) ;
- **gamme** : Ensemble des combinaisons jouées par un autre instrument sur le **motif** (réparti sur 1 voix) ;

— **forme rythmique : motif + gamme**

Les **motifs** sont fixes, contrairement aux **gammes** regroupent l'ensemble des possibilités qui peuvent être rencontrées en situation réelle (sur une partition ou lors d'une performance de batterie).

Un **motif** détermine la signature rythmique d'une **forme rythmique**, et donc de sa **gamme**. Il détermine aussi avec quel instrument unique sera jouée la **gamme** et comment tous les instruments de la **forme rythmique** se répartiront en voix.

Les **formes rythmiques** devront être distribuées dans 4 grandes catégories<sup>8</sup> [28] :

FR	SR	Subdivisions	Possibles	nb voix
binaires	simple	doubles-croches	triolet, sextolet	2
jazz	simple	triolet	croches et doubles	2
ternaires	complexe	croches	duolet, quartelet	2
afros-cubains	simple	croches	-	3

TABLE 3.3 – Systèmes

Nous exposerons 3 **formes rythmiques** afin d'illustrer les propos de cette section :

- 4/4 binaire
- 4/4 jazz
- 4/4 afro-cubain

Le **motif** et la

### Détection de la signature rythmique

La partie **motif** des **FR** sera utilisée pour la **définition des signature rythmiques**. La détection de la signature rythmique est importante, non seulement pour connaître le nombre de temps par mesure ainsi que le nombre de subdivisions pour chacun de ces temps, mais aussi pour savoir comment écrire l'unité de temps et ses subdivisions.

La figure 3.15 montre deux signatures rythmiques différentes. L'une (exemple 1) est *simple* (2 temps binaires sur lesquels sont joués des triolets), l'autre (exemple 2) est *complexe* (2 temps ternaires). Le jazz est traditionnellement écrit en binaire avec ou sans triolet (même si cette musique est dite ternaire alors que le rock ternaire sera plutôt écrit comme dans l'exemple 2).

8. SR du tableau signifie « signature rythmique »



Exemple 1



Exemple 2

FIGURE 3.15 – signature rythmique

### Choix d'une grammaire

Il faut prendre en compte l'existence potentielle de plusieurs grammaires qui regroupent les contenus MIDI par signature rythmique. Le choix d'une grammaire pondérée doit être fait avant le parsing puisque qparse prend en entrée un fichier MIDI et un fichier wta (grammaire). C'est pour cette raison que la signature rythmique doit être définie avant le choix de la grammaire. Il faudrait les trouver automatiquement sans autres indications que les contenus MIDI. Par conséquent, les **motifs** des **formes rythmiques** devront être recherchés sur l'input (fichiers MIDI) avant le lancement du parsing, afin de déterminer la signature rythmique en amont. Cette tâche devra probablement être effectuée par utilisation d'apprentissage automatique.

### Séparation des voix

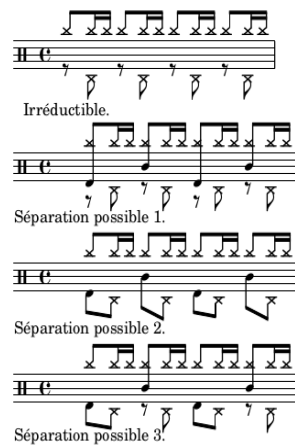
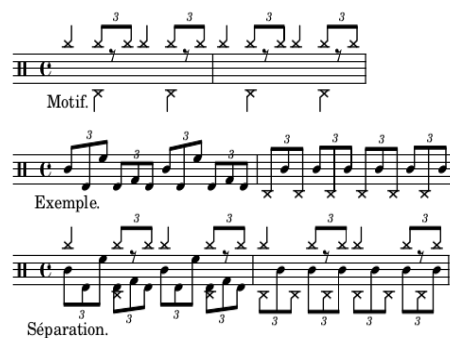
L'objectif de la procédure : ?

Le principe de la procédure : ?

Quelles sont les difficultés et les choix de séparation de voix et comment un algorithme pourrait faire ces choix ?

Ici, la **forme rythmique** est construite sur un modèle rock avec une signature rythmique en 4/4.

La première ligne de la figure 3.16 est appelée « Irréductible » car il n'y a pas d'autre choix de séparation des voix pour la ride et le charley au pied. La troisième séparation proposée est privilégiée car elle répartit selon deux voix, une voix pour les mains (ride et caisse claire) et une voix pour les pieds (charley et grosse caisse). Ce choix paraît plus équilibré car deux instruments sont utilisés par voix (contrairement séparations possibles 1 et 2 de la figure 3.16) et plus logique pour le lecteur puisque les mains sont en haut et les pieds en bas.

FIGURE 3.16 – **motif** 4-4 binaireFIGURE 3.17 – **Motif** 4-4 jazz

Dans la plupart des méthodes, le charley n'est pas écrit car il est considéré comme évident en jazz traditionnel. Ici, le parti pris est de tout écrire.

Dans la figure 3.17, les mesures 1 et 2 de la ligne « Exemple » combinées avec le **motif** de la première ligne, sont des cas typiques de la batterie jazz. Tout mettre sur la voix haute serait surchargé. De plus, la grosse caisse entre très souvent dans le flot des combinaisons de toms et de caisse claire et son écriture séparée serait inutilement compliquée et peu intuitive pour le lecteur. Le choix de séparation sera donc de laisser les cymbales jouées à la main en haut, et les toms, la caisse claire, la grosse caisse et la pédale de charley en bas.

La figure 3.18 montre un exemple minimaliste de **forme rythmique** afro-latin [28].

Cette **forme rythmique** doit être écrite sur trois voix car la voix centrale est souvent plus complexe que sur la figure et la mélanger avec le haut ou le bas serait surchargé et peu lisible.

FIGURE 3.18 – **Forme rythmique** 4-4 afro-latin

### Simplification de l'écriture

Les règles de **simplification** (les combinaisons de réécritures) seront extraites des voix séparées des **formes rythmiques**. Les explications qui suivent seront appuyées par une réécriture guidée dans la section 4.4.

Les **gammes** qui accompagnent les **motifs** étayent toutes les combinaisons d'un **FR** et elles permettent, combinées avec le **motif**, de définir ses propres règles de simplification.

Voici les différentes étapes à suivre :

- pour chaque **gamme** d'une **forme rythmique**, faire un arbre de rythme représentant la **gamme** combinée avec le **motif** de la **FR** ;
- pour chaque arbre de rythmes obtenus, séparer les voix et faire un arbre de rythme par voix ;
- pour chaque voix (arbres de rythmes) obtenus, extraire tous les nœuds qui nécessitent une simplification et écrire la règle.

Certaines précisions concernant l'extraction de ces règles sont nécessaires. Il s'agit de précisions à propos de la durée, des silences et de la présence ou non d'ouvertures de charley dans les instruments joués. Nous avons discuté de ces problèmes plus haut dans ce chapitre.

Voici quelques règles inhérentes à la simplification de l'écriture pour la batterie :

Même si on favorise l'usage des silences pour l'écart entre les notes n'appartenant pas au même temps, on les remplace systématiquement par un point pour 2 notes au sein d'un même temps.

Dans la figure 3.19, les « suites » sont des notations possibles relatives aux arbres 1 ou 2.

*Rappel :*

*cf* = charley fermé joué à la main ;

*co* = charley ouvert joué à la main ;

*pf* = charley fermé joué au pied.

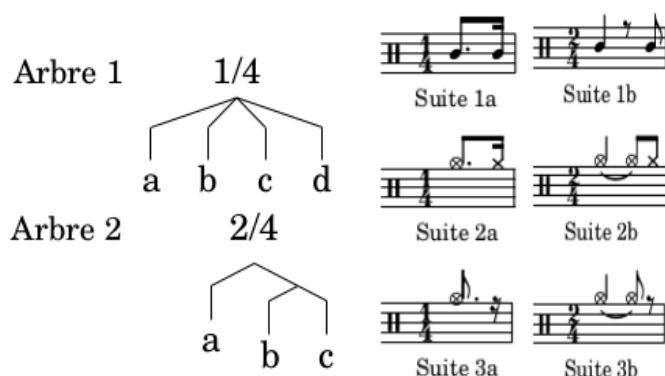


FIGURE 3.19 – Simplifications — arbres et notations possibles

Soit l'arbre 1 de la figure 3.19 dans lequel :

- a et d sont des instruments de la batterie (x);
- b et c sont des continuations (t).

Pour chacune des conditions suivantes, une suite de la figure 3.19 est attribuée :

- Si a n'est pas un co :  
⇒ Suite 1a.
- Si a est un co :
  - Si d est un cf :  
⇒ Suite 2a.
  - Si d est un pf :  
⇒ Suite 3a : d devient un silence (r).

Soit l'arbre 2 de la figure 3.19 dans lequel :

a et c sont des instruments de la batterie (x);  
b est une continuation (t); Pour chacune des conditions suivantes, une suite de la figure 3.19 est attribuée :

- Si a n'est pas un co :  
⇒ Suite 1b, b devient un silence.
- Si a est un co :
  - Si c est un cf :  
⇒ Suite 2b, b devient une liaison et c devient un cf.
  - Si c est un pf :  
⇒ Suite 3b : b devient une liaison et c devient un silence.

## Conclusion

Dans ce chapitre, nous avons formalisé une notation de la batterie inspirée des méthodes de batterie Dante Agostini, modélisé cette notation pour la transcription de données MIDI en partition. Nous avons ensuite parlé

de l'outil utilisé pour les transcriptions manuelles en mettant en avant que cet outil devrait être utilisé pour la transcription de la batterie et en précisant que tous les codes lilypond pour la création des figures et partition de ce mémoire sont en accès libre sur github. Nous avons ensuite décrit qparse qui l'outil que le travail de ce mémoire cherche à améliorer. Enfin, nous avons exposé une approche de type dictionnaire (les « formes rythmiques ») pour détecter une signature rythmique, choisir une grammaire pondérée appropriée et énoncer des règles de séparation des voix et de simplification de l'écriture.



## EXPÉRIMENTATIONS

### Sommaire

---

4.1	Le jeu de données . . . . .	50
4.2	Analyses et transcriptions manuelles . . . . .	51
4.3	Transcription polyphonique par parsing . . . . .	55
4.4	Réécriture guidée par une forme rythmique . . . . .	55
4.5	BILAN : résultats — évaluation — discussion . . . . .	60

---

### Introduction

Dans ce chapitre, nous présenterons le jeu de données et les analyse MIDI-Audio et transcriptions manuelles.

Problématique :

choix d'un outil de transcription ?

passage au polyphonique indispensable pour la suite du travail et pour l'usage des formes rythmiques.

Finir la chaîne de traitement indispensable pour obtenir des résultats chiffrés possible à évaluer.

Nous présenterons mes trois contributions principales :

- le code lilypond normalisé pour la transcription de la batterie avec la notation de type agostini.
- les différentes étapes de résolution du passage au polyphonique.
- l'expérimentation d'une forme rythmique implémentable qui devra être utilisé comme base de connaissances pour augmenter la rapidité et la qualité en sortie de Qparse et comme une méthode de création de nouvelles formes rythmiques.

Enfin, nous finirons par une discussion sur les avancées réalisées dans ce travail, la pertinence des choix qui ont été faits et les moyens d'évaluer les résultats potentiels.

## 4.1 Le jeu de données

Nous avons utilisé le Groove MIDI Dataset<sup>1</sup> [34] (GMD) qui est un jeu de données mis à disposition par Google sous la licence Creative Commons Attribution 4.0 International (CC BY 4.0).

Le GMD est composé de 13,6 heures de batterie sous forme de fichiers MIDI et audio alignés. Il contient 1150 fichiers MIDI et plus de 22 000 mesures de batterie dans les styles les plus courants et avec différentes qualités de jeu. Tout le contenu a été joué par des humains sur la batterie électronique Roland TD-11 (figure 4.1).

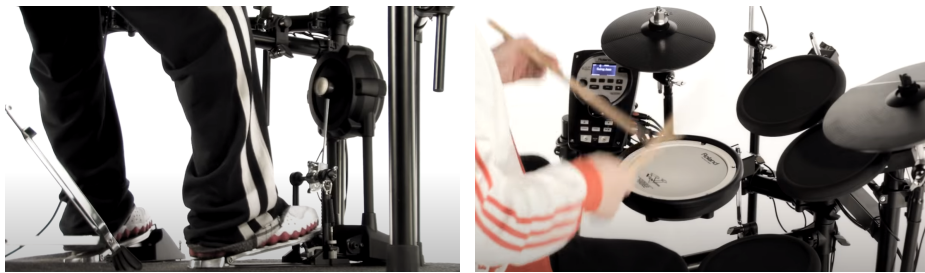


FIGURE 4.1 – Batterie électronique

Source : [https://www.youtube.com/watch?v=BX1V\\_IE0g2c](https://www.youtube.com/watch?v=BX1V_IE0g2c)

Autres critères spécifiques au GMD :

- Toutes les performances ont été jouées au métronome et à un tempo choisi par le batteur.
- 80% de la durée du GMD a été joué par des batteurs professionnels qui ont pu improviser dans un large éventail de styles. Les données sont donc diversifiées en termes de styles et de qualités de jeu (professionnel ou amateur).
- Les batteurs avaient pour instruction de jouer des séquences de plusieurs minutes ainsi que des fills<sup>2</sup>
- Chaque performance est annotée d'un style (fourni par le batteur), d'une signature rythmique et d'un tempo ainsi que d'une identification anonyme du batteur.
- Il a été demandé à 4 batteurs d'enregistrer le même groupe de 10 rythmes dans leurs styles respectifs. Ils sont dans les dossiers évaluation du GMD.
- Les sorties audio synthétisées ont été alignées à 2 ms près sur leur fichier MIDI.

1. <https://magenta.tensorflow.org/datasets/groove>

2. Un *fill* est une séquence de relance dont la durée dépasse rarement 2 mesures. Il est souvent joué à la fin d'un cycle pour annoncer le suivant.

### Format des données

Le Roland TD-11 enregistre les données dans des fichiers MIDI et les divise en plusieurs pistes distinctes :

- une pour le tempo et l'indication de mesure ;
- une pour les changements de contrôle (position de la pédale de charley) ;
- une pour les notes.

Les changements de contrôle sont placés sur le canal 0 et les notes sur le canal 9 (qui est le canal canonique pour la batterie).

Pour simplifier le traitement de ces données, ces trois pistes ont été fusionnées en une seule piste qui a été mise sur le canal 9.

## 4.2 Analyses et transcriptions manuelles

Ces analyses ont été faites dans le cadre de transcriptions manuelles à partir de fichiers MIDI et Audio du GMD.

### Comparaisons de transcriptions

Pour les comparaisons de transcriptions, les transcriptions manuelles (TM) ont été éditées à l'aide de Lilypond<sup>3</sup> ou MuseScore<sup>4</sup> et les transcriptions automatiques (TA) ont toutes été générées par import d'un fichier MIDI dans MuseScore.

#### Exemple d'analyse 1

Transcription manuelle ⇒ Transcription automatique



- Erreur d'indication de mesure (3/4 au lieu de 4/4) ;
- Les silences de la mesure 1 de la TA sont inutilement surchargés ;
- La noire du temps 4 de la mesure 1 de la TM est devenue les deux premières notes (une double-croche et une croche) d'un triolet sur le temps 1 de la mesure 2 de la TA.

Transcription manuelle  $\Rightarrow$  Transcription automatique



### Exemple d'analyse 2

- Les doubles croches ont été interprétées en quintolet
- La deuxième double-croche est devenue une croche.

### Exemple d'analyse 3

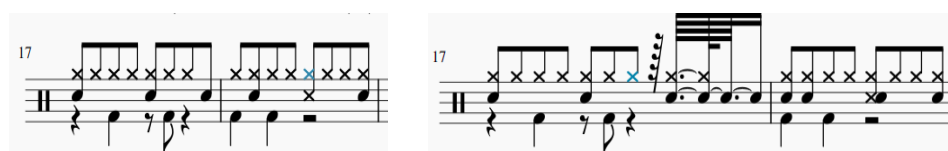
Transcription manuelle  $\Rightarrow$  Transcription automatique



- Les grosses-caisses, les charleys et les caisses-claires ont été décalés d'un temps vers la droite.
- Les toms basses des temps 1 et 2 de la mesure 2 de la TM ont été décalés d'une double croche vers la droite dans la TA.
- La première caisse-claire de la mesure 1 devient binaire dans la TA alors qu'elle appartenait à un triolet dans la TM.
- Le triolet de tom-basse du temps 4 de la mesure 2 de la TA n'existe pas la TM.

### Exemple d'analyse 4

Transcription manuelle  $\Rightarrow$  Transcription automatique



Sur le temps 4 de la mesure 1, la deuxième croche a été transcrite d'une manière excessivement complexe !

3. <http://lilypond.org/>

4. <https://musescore.com/>

**Exemple d'analyse 5 (flas)**

Transcription manuelle



Transcription automatique



- Le premier fla est reconnu comme étant un triolet contenant une quadruple croche suivie d'une triple croche au lieu d'une seule note ornementée.
- Le deuxième fla est reconnu comme étant un accord.
- Les deux double en contre-temps sur le temps 4 de la TM sont mal quantifiée dans la TA.
- La TA ne reconnaît qu'une mesure quand la TM en transcrit deux. En effet, la TA a divisé par deux la durée des notes afin de les faire tenir dans une mesure à 4 temps dont les unités de temps sont les noires. Par exemple, le soupir du temps 2 de la TM devient un demi-soupir sur le contre-temps du temps 1 dans la TA. Ou encore, la noire (pf, voir le tableau 3.2) sur le temps 1 de la mesure 2 de la TM suivie d'un demi-soupir devient une croche pointée sur le temps 3 de la TA.
- Autre problème : certaines têtes de notes sont mal attribuées. Par exemple, le charley ouvert en contre-temps sur le temps 2 de la mesure 2 de la TM devrait avoir le même symbole sur la TA. Idem pour les cross-sticks.

**Conclusion d'analyse**

Ces analyses ont montré la difficulté pour un logiciel comme MuseScore d'offrir une partition lisible. Les raisons sont le fait que les fichiers MIDI ne sont pas encore quantifiés mais aussi qu'il n'y a pas de reconnaissance de la forme du rythme impliquant sa position dans la mesure. Cette reconnaissance pourrait permettre de rectifier les problèmes de signature rythmique ainsi que les problèmes de décalage de temps. La reconnaissance de la forme du rythme permettrait aussi de supprimer les aberrations du type de celle de l'exemple d'analyse 4, puisque l'erreur sur cet exemple

serait reconnue comme un élément qui ne rentre pas dans le cadre de la forme de rythme en question. La dernière raison qui rend le travail difficile est l'identification des flas, comment savoir si deux notes jouées très proches sont :

- séparées et rapides,
- mal jouées à l'unisson (accord),
- ou forment un fla ?

### Transcription de partition

La figure 3.11 est la transcription manuelle des fichiers *004\_jazz-funk\_116\_beat\_4-4.mid* et *004\_jazz-funk\_116\_beat\_4-4.wav* du GMD.

Cette transcription a été entièrement faite avec Lilypond (voir le code lilypond sur le git [https://github.com/MartinDigard/Stage\\_M2\\_Inria](https://github.com/MartinDigard/Stage_M2_Inria)). Il s'agit d'une partition d'un 4/4 binaire dont le fichier MIDI est annoncé dans le GMD de style «jazz-funk» probablement en raison de la ride de type shabada rapide (le ternaire devient binaire avec la vitesse) combiné avec l'after-beat de type rock (caisse-claire sur les deux et quatre).

La transcription manuelle de la partition de la figure 3.11 et l'analyse d'autres fichiers MIDI (voir section 4.2) m'ont mené aux observations suivantes :

- Vitesse inférieure à 40 : ghost-note ;
- Vitesse supérieure à 90 : accent ;
- Pas d'intention d'accent ni de ghost-note pour une vitesse entre 40 et 89 ;
- Les accents et les ghost-notes ne sont significatifs ni pour les instruments joués au pied, ni pour les cymbales crash.  
En effet, certaines vitesses en dessous de 40 étant détectées et inscrites dans les données MIDI sont dues au mouvement du talon du batteur qui bat la pulsation sans particulièrement jouer le charley. Ce mouvement est perçu par le capteur de la batterie électronique mais le charley n'est pas joué.
- Au final, j'ai relevé les ghost-notes et les accents pour la caisse claire ainsi que les accents pour les toms et les cymbales rythmiques (charley et ride).

### Conclusion sur les transcriptions manuelles

La transcription des données audio et MIDI contenues dans ces fichiers a permis une analyse plus approfondie des critères à relever pour chaque événement MIDI et de la manière de les considérer dans un objectif de transcription en partition lisible pour un musicien (Voir la section 3.3).

### 4.3 Transcription polyphonique par parsing

Les Jams permettent de passer du monophonique au polyphonique. On regroupe tous les événements qu'on estime être à des dates suffisamment proches dans des clusters que nous avons appelés « Jam ». Un Jam représente une séquence de points successifs dans un segment d'entrée avec des informations supplémentaires sur leur rôle respectif, en supposant qu'ils sont tous alignés à la même date lors de la quantification. Un Jam est caractérisé par l'index de son premier point dans le segment d'entrée et sa longueur (nombre de points). Une fois dans les Jams, les événements sont identifiés selon les critères suivants :

- Ignored, les offsets sont ignorés ;
- Note, si c'est une note simple ;
- Rest, si c'est un silence ;
- GraceNote, si c'est un fla ;
- Error.

### 4.4 Réécriture guidée par une forme rythmique

La démonstration qui suit est basée sur la partition de référence de la figure 3.11 puisque la forme rythmique qui sera utilisée en est directement extraite.

Nous allons montrer :

- la composition de cette forme rythmique ;
- son état finale, c'est à dire toutes les combinaisons entièrement écrites en notation correcte sur partition ;
  - ⇒ cela constituera une référence pour la réécriture ;
- un exemple de transformation de la forme rythmique en arbre de rythme ;
- l'application de la séparation des voix sur cet exemple basé sur la référence citée précédemment (la forme rythmique en question) ;
  - ⇒ l'arbre de départ sera alors séparé en autant d'arbres qu'il y a de voix (deux arbres pour cette forme rythmique) ;
- les règles de simplification propres à la forme rythmique dont nous parlons.

L'objectif de cette démonstration est de montrer comment un jeu de plusieurs formes rythmiques pourrait être implémenter dans le cadre d'une approche dictionnaire.



FIGURE 4.2 – Motifs et gammes

## Motifs et gammes

### Motifs

À partir de la partition de référence, les deux motifs de la figure 4.2 peuvent être systématisés. Le motif 1 est joué du début jusqu'à la mesure 18 avec des variations et des fills et le motif 2 est joué de la mesure 23 à la mesure 28 avec des variations. Ces deux motifs sont très classiques et pourront être détectés dans de nombreuses performances.

### Gammes

Les gammes de la figure 4.2 étayent toutes les combinaisons d'un motif en 4/4 binaires jusqu'aux doubles croches.

Les lignes 1 et 2 traitent les croches. La ligne 1 a 2 mesures dont la première ne contient que des noires et la deuxième que des croches en contre-temps. Ces deux possibilités sont combinées de manière circulaire dans les 3 mesures de la deuxième ligne.

Les lignes 3, 4 et 5 traitent les doubles-croches. La ligne 3 a 2 mesures dont la première ne contient que des croches et la deuxième que des doubles-croches en contre-temps. Ces deux possibilités sont combinées de manière circulaire dans les lignes 4 et 5 qui contiennent chacune 3 mesures.

## Formes rythmiques — motifs et gammes combinés

Pour la suite de cette démonstration, je utiliserai le motif 1 de la figure 4.2. à commenter un peu plus, notamment pour dire si la combinaison est faite automatiquement ou non



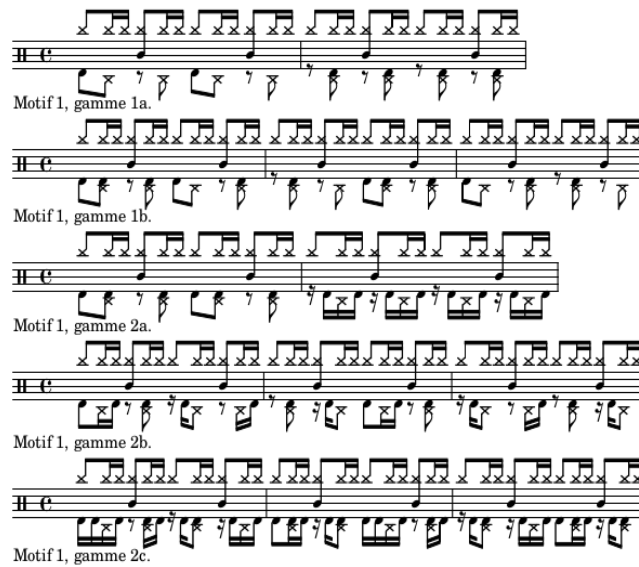


FIGURE 4.3 – Partition d'un forme rythmique en 4/4 binaire

### Représentation de la forme rythmique en arbres de rythmes

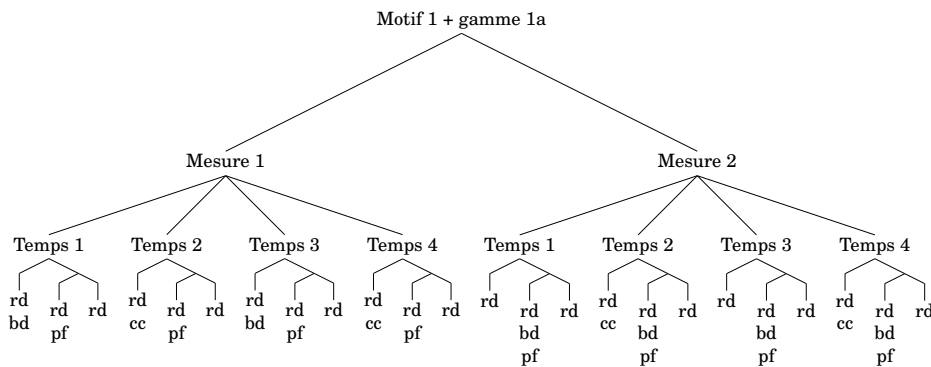


FIGURE 4.4 – Arbre de rythme — forme rythmique

L'arbre de la figure 4.4 servira de base pour le suite de l'expérimentation. Comme indiqué à la racine de l'arbre, il représente la première ligne de la figure 4.3. Même si cet arbre représente parfaitement le rythme concerné, il manque des indications de notation telles que les voix spécifiques à chaque partie du rythme ainsi que les choix d'écriture pour les distances qui séparent les notes de chaque voix entre elles en termes de durée.

## Réécriture — séparation des voix et simplification

### La séparation des voix

Ainsi l'arbre syntaxique de départ est divisé en autant d'instruments qui le constituent et les voix seront regroupées en suivant les règles du forme rythmique.

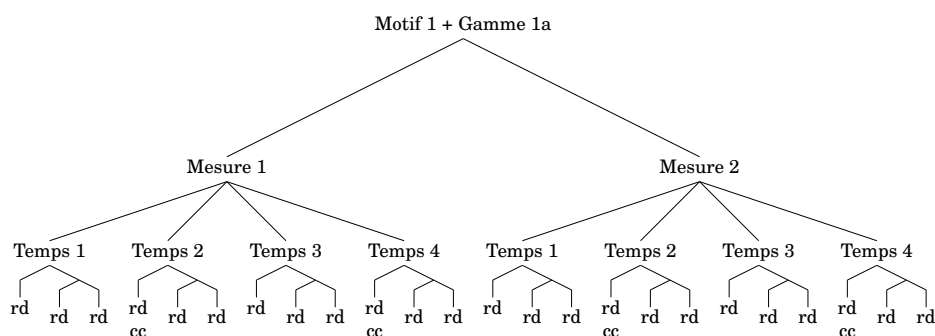


FIGURE 4.5 – Arbres de rythme — voix haute

La voix haute (figure 4.5) regroupe la ride et la caisse-claire sur les ligatures du haut.

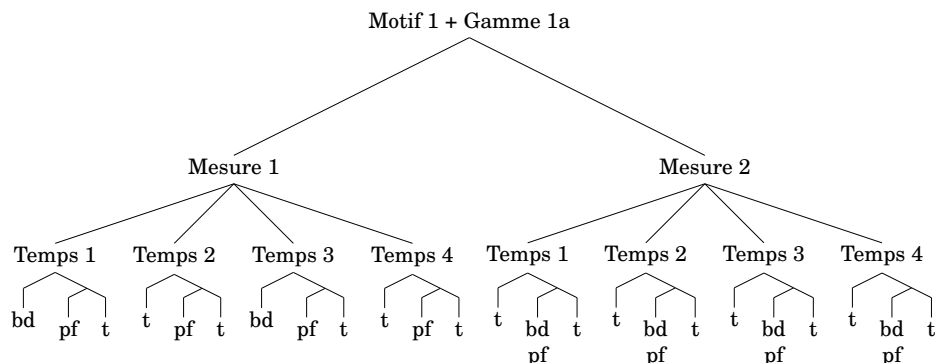


FIGURE 4.6 – Arbres de rythme — voix basse

La voix basse (figure 4.6) regroupe la grosse-caisse et le charley au pied sur les ligatures du bas.

### Les règles de simplifications

L'objectif des règles de simplifications est de réécrire les écarts de durées qui séparent les notes d'une manière appropriée pour la batterie et qui soit la plus simple possible. Les ligatures relient les notes d'un temps entre elles afin de rendre la pulsation visuelle).

Pour les figures ci-dessous :

- x = une note ;
- r = un silence ;
- t = une continuation (point ou liaison)

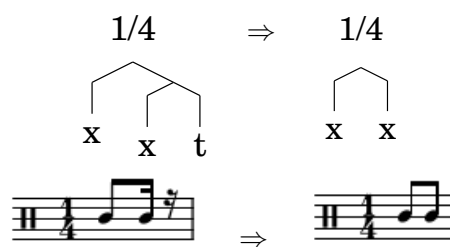


FIGURE 4.7

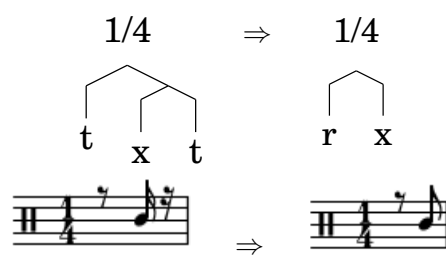


FIGURE 4.8

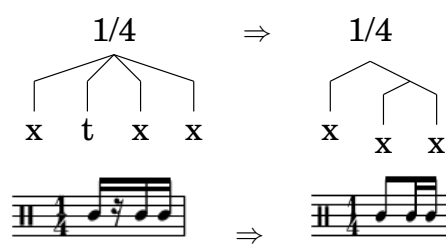


FIGURE 4.9

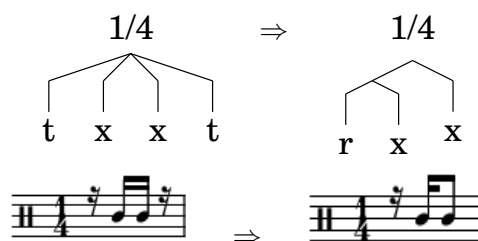


FIGURE 4.10

Ces règles ont été tirées de l'ensemble des arbres de la forme rythmique. Les arbres manquants seront mis en annexe.

Les règles remplacent par un silence les continuations (t) qui sont au début d'un temps. Cela est valable pour cette forme rythmique mais lorsqu'il y a des ouvertures de charley, cela n'est pas toujours applicable.

### Conclusion sur cette réécriture guidée

La méthode des formes rythmiques étant basée sur une approche dictionnaire, Le premier objectif de cette réécriture guidée est d'orienter la recherche d'autres formes rythmiques par observation du jeu de données et de montrer comment les construire pour agrandir la base de connaissance de Qparse pour la transcription de la batterie.

## 4.5 BILAN : résultats — évaluation — discussion

Cette section regroupe les avancées qui ont été réalisées par rapport aux objectifs de départ ainsi qu'une réflexion sur le moyen d'évaluer les résultats de l'ADT avec Qparse. Nous avons amélioré le système de quantification de Qparse pour la batterie, notamment le passage à la polyphonie avec les Jams.

Nous avons pu obtenir des arbres de parsing corrects en améliorant les grammaires avec des fichiers MIDI courts.

Puis, une sortie MEI a aussi été obtenue (encore à vérifier).

Dans cette section, nous discuterons sur la pertinence de l'ensemble des choix qui ont été faits. Nous ferons un bilan des différentes avancées qui ont été faites ou non et nous tenterons d'en expliquer la ou les raisons.

- Le choix de travailler avec Lilypond et non Verovio. Ce choix était motivé par la liberté totale concernant la notation de la batterie dont un et la disponibilité d'un set de notation de type Agostini. C'est la seule application qui me permettait d'écrire la notation de la batterie exactement comme je le souhaitais.
- Avancé de la chaîne de traitement (nous sommes arrivé aux arbres de parsing, nous avons traité le polyphonique (identification des

regroupements de notes<sup>5)</sup> ⇒ Quelques arbres ont été obtenus sur des exemples simples (<sup>6)</sup>)

— 2 dimensions de le travail fourni :

- La volonté de pousser un exemple simple jusqu'au bout de la chaîne pour obtenir des résultats et une évaluation sur au moins un exemple ; - La réalité du travail à fournir pour faire avancer sur la chaîne de traitement. ⇒ Une solution aurait été de considérer les arbres de parsing obtenus après le traitement du polyphonique comme un résultat local possible à évaluer au lieu d'attendre que la chaîne arrive jusqu'à la génération d'une partition mais cela n'était pas prioritaire pendant le stage.

— Création d'un jeu de forme rythmique basique représentatif des différents styles à recouvrir. Ce jeu n'a pas pu être créé, car comme vu plus haut, je me suis focalisé sur un exemple pour pouvoir le vérifier entièrement et dans l'espoir de pouvoir le tester en fin de chaîne. **Évaluation** Matcher les motifs aurait été indispensable pour obtenir une quantité de résultats qui justifieraient une évaluation automatique permettant de faire des graphiques.

L'évaluation fut entièrement manuelle car :

⇒ Très dure automatiquement : il faut comparer 2 partitions (réf VS output) Pour l'évaluation, il aurait fallu produire un module.  
<dam>je ne sais pas si tu auras encore le temps de faire ça, sinon il faudra décrire comment tu aurais aimé évaluer, proprement et sans résultats chiffrés</dam> L'évaluation est-elle automatique ou manuelle?

Possibilité d'un export lilypond en arbre pour comparer l'output avec la transcription manuelle.

Possibilité de transformer lilypond(output) et lilypond(ref) en ScoreModel ou MEI pour les comparer et faire des statistiques. Si transformés en MEI : diffscore de Francesco. Possibilité de transformer lilypond(output) et lilypond(ref) en MusicXML pour les comparer ou dans Music21. L'expérimentation peut-être considérer comme une évaluation manuelle? (magicien d'Oz)

Lilypond vers MIDI + output vers MIDI ⇒ Comparaison des MIDI dumpés.

La transcription automatique de la batterie est un sujet passionnant mais difficile : Obtenir la totalité des éléments nécessaires pour le mémoire nécessiterait plus de temps. Une base solide spécifique à la batterie a néanmoins été générée. Elle sera un bon point de départ pour les travaux futurs dont plusieurs propositions sont énoncés dans le présent document.

5. fla ou accords entre autres...

6. exemple de 2 mesures, voir ...



## CONCLUSION GÉNÉRALE

Dans ce mémoire, nous avons traité de la problématique de la transcription automatique de la batterie. Son objectif était de transcrire, à partir de leur représentation symbolique MIDI, des performances de batteur de différents niveaux et dans différents styles en partitions écrites.

Nous avons avancé sur le parsing des données MIDI établissant un processus de regroupement des événements MIDI qui nous a permis de faire la transition du monophonique vers le polyphonique. Une des données importante de ce processus était de différencier les nature des notes d'un *accord*, notamment de distinguer lorsque 2 notes constituent un *accord* ou un *fla*.

Nous avons établis des *grammaires pondérées* pour le parsing qui correspondent respectivement à des métriques spécifiques. Celles-ci étant sélectionnables en amont du parsing, soit par indication des noms des fichiers MIDI, soit par reconnaissance de la métrique avec une approche dictionnaire de patterns prédéfinis <sup>7</sup> qu'il serait pertinent de mettre en œuvre en machine learning.

Nous avons démontré que l'usage des *systèmes* élimine un grand nombre de calcul lors de la réécriture. Pour la séparation des voix grâce au motif d'un système et pour la simplification grâce aux gammes du motif d'un système. Nous avons aussi montré comment, dans des travaux futurs, un système dont le motif serait reconnu en amont dans un fichier MIDI pourrait prédéfinir le choix d'une grammaire par la reconnaissance d'une métrique et ainsi améliorer le parsing et accélérer les choix ultérieurs dans la chaîne de traitement en terme de réécriture.

Il sera également intéressant d'étudier comment l'utilisation de LM peut améliorer les résultats de l'AM, voir [2], et ouvrir la voie à la génération entièrement automatisée de partitions de batterie et au problème général de l'AMT de bout en bout.[11]

---

7. *Motifs* dans les *systèmes* de la présente proposition.





## BIBLIOGRAPHIE

- [1] A. Danhauser. *Théorie de la musique*. Edition Henry Lemoine, 41 rue Bayen - 75017 Paris, Édition revue et augmentée - 1996 edition, 1996. – Cité pages 7, 16, 17 et 33.
- [2] H. C. Longuet-Higgins. Perception of melodies. 1976. – Cité pages 11 et 15.
- [3] Meinard Müller. *Fundamentals of Music Processing*. 01 2015. – Cité page 12.
- [4] Gaël Richard et al. De fourier à la reconnaissance musicale. Available at <https://interstices.info/de-fourier-a-la-reconnaissance-musicale/> (2019/02/15). – Cité page 12.
- [5] Caroline Traube. Quelle place pour la science au sein de la musicologie aujourd’hui? *Circuit*, 24(2) :41–49, 2014. – Cité page 12.
- [6] Leonard Bernstein Office. The unanswered question : Six talks at harvard. Available at <https://leonardbernstein.com/about/educator/norton-lectures> (2021/01/01). – Cité page 12.
- [7] Bénédicte Poulin-Charronnat and Pierre Perruchet. Les interactions entre les traitements de la musique et du langage. *La Lettre des Neurosciences*, 58 :24–26, 2018. – Cité page 13.
- [8] Mikaela Keller, Kamil Akesbi, Lorenzo Moreira, and Louis Bigo. Techniques de traitement automatique du langage naturel appliquées aux représentations symboliques musicales. In *JIM 2021 - Journées d’Informatique Musicale*, Virtual, France, July 2021. – Cité page 13.
- [9] Peter Wunderli. Ferdinand de saussure : La sémiologie et les sémiologies. *Semiotica*, 2017(217) :135–146, 2017. – Cité page 13.
- [10] Junyan Jiang, Gus Xia, and Taylor Berg-Kirkpatrick. Discovering music relations with sequential attention. In *NLP4MUSA*, 2020. – Cité page 13.
- [11] Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Holger Kirchhoff, and Anssi Klapuri. Automatic music transcription : Chal-

- lenges and future directions. *Journal of Intelligent Information Systems*, 41, 12 2013. – Cité pages 14, 15, 21 et 63.
- [12] Daniel Harasim, Christoph Finkensiep, Petter Ericson, Timothy J O'Donnell, and Martin Rohrmeier. The jazz harmony treebank. – Cité pages 4, 14 et 25.
- [13] Georges Paczynski. *Une histoire de la batterie de jazz*. OUTRE MESURE, 1997. – Cité page 15.
- [14] Chih-Wei Wu, Christian Dittmar, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Meinard Müller, and Alexander Lerch. A review of automatic drum transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9) :1457–1483, 2018. – Cité pages 15, 22 et 25.
- [15] Elaine Gould. *Behind bars : the definitive guide to music notation*. Faber Music Ltd, 2016. – Cité page 16.
- [16] Kentaro Shibata, Eita Nakamura, and Kazuyoshi Yoshii. Non-local musical statistics as guides for audio-to-score piano transcription. *Information Sciences*, 566 :262–280, 2021. – Cité pages 17, 23 et 24.
- [17] Nicolas Guiomard-Kagan. *Traitement de la polyphonie pour l'analyse informatique de partitions musicales*. PhD thesis. – Cité page 17.
- [18] Moshekwa Malatji. Automatic music transcription for two instruments based variable q-transform and deep learning methods, 10 2020. – Cité page 22.
- [19] Antti J. Eronen. Musical instrument recognition using ica-based transform of features and discriminatively trained hmms. *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, 2 :133–136 vol.2, 2003. – Cité page 22.
- [20] Hiroshi G. Okuno Kazuyoshi Yoshii, Masataka Goto. Automatic drum sound description for real-world music using template adaptation and matching methods. *International Conference on Music Information Retrieval (ISMIR)*, pages 184–191, 2004. – Cité page 22.
- [21] Francesco Foscarin, Florent Jacquemard, Philippe Rigaux, and Masahiko Sakai. A Parse-based Framework for Coupled Rhythm Quantization and Score Structuring. In *MCM 2019 - Mathematics and Computation in Music*, volume Lecture Notes in Computer Science of *Proceedings of the Seventh International Conference on Mathematics and Computation in Music (MCM 2019)*, Madrid, Spain, June 2019. Springer. – Cité pages 23 et 25.
- [22] C. Agon, K. Haddad, and G. Assayag. Representation and rendering of rhythm structures. In *Proceedings of the First International*

- Symposium on Cyber Worlds (CW'02)*, CW '02, page 109, USA, 2002. IEEE Computer Society. – Cité page 24.
- [23] Florent Jacquemard, Pierre Donat-Bouillud, and Jean Bresson. A Term Rewriting Based Structural Theory of Rhythm Notation. Research report, ANR-13-JS02-0004-01 - EFFICACe, March 2015. – Cité page 24.
- [24] Florent Jacquemard, Adrien Ycart, and Masahiko Sakai. Generating equivalent rhythmic notations based on rhythm tree languages. In *Third International Conference on Technologies for Music Notation and Representation (TENOR)*, Coruña, Spain, May 2017. Helena Lopez Palma and Mike Solomon. – Cité page 25.
- [25] R. Marxer and J. Janer. Study of regularizations and constraints in nmf-based drums monaural separation. In *International Conference on Digital Audio Effects Conference (DAFx-13)*, Maynooth, Ireland, 02/09/2013 2013. – Cité page 25.
- [26] J.-F. Juskowiak. *Rythmiques binaires 2*. Alphonse Leduc, Editions Musicales, 175, rue Saint-Honoré, 75040 Paris, 1989. – Cité page 27.
- [27] Dante Agostini. *Méthode de batterie, Vol. 3*. Dante Agostini, 21, rue Jean Anouilh, 77330 Ozoir-la-Ferrière, 1977. – Cité page 27.
- [28] O. Lacau J.-F. Juskowiak. *Systèmes drums n. 2*. MusicCom publications, Editions Joseph BÉHAR, 61, rue du Bois des Joncs Marins - 94120 Fontenay-sous-Bois, 2000. – Cité pages 30, 43 et 45.
- [29] Frédéric Canet. La batterie... mot à mot! Available at <https://rimshotetghostnote.fr/> (2021). – Cité page 34.
- [30] M. Laurson. Patchwork : a visual programming language and some musical applications. 1996. – Cité page 38.
- [31] Jean Bresson, Carlos Agon, and Gérard Assayag. Openmusic visual programming environment for music composition, analysis and research. – Cité page 38.
- [32] Dick Grune and Cerial JH Jacobs. Parsing techniques. *Monographs in Computer Science*. Springer,, page 13, 2007. – Cité page 39.
- [33] Manfred Droste, Werner Kuich, and Heiko Vogler. *Handbook of weighted automata*. Springer Science & Business Media, 2009. – Cité page 39.
- [34] Jon Gillick, Adam Roberts, Jesse Engel, Douglas Eck, and David Bamman. Learning to groove with inverse sequence transformations. In *International Conference on Machine Learning (ICML)*, 2019. – Cité page 50.

