Institut National des Langues et Civilisations Orientales

Département Textes, Informatique, Multilinguisme

Titre du mémoire

MASTER

TRAITEMENT AUTOMATIQUE DES LANGUES

Parcours :

Ingénierie Multilingue

par

Martin DIGARD

Directeur de mémoire :

Damien NOUVEL

Encadrant :

Florent JACQUEMARD

Année universitaire 2020/2021

Table des matières

Liste des figures

Liste des tableaux

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 d'amélioration de Qparse, un outil de transcription et d'écriture automatique de la musique sur sa capacité à transcrire la batterie. Nous ne parlerons donc pas directement de langues naturelles, mais de l'écriture automatique de partitions de musique à partir de données audios. Cette exercice nécessitera la manipulation d'un langage musical codifié avec une grammaire (solfège, durées, nuances, volumes) et soulèvera des problématiques concernées par les techniques du traitement automatique des langues.

La batterie est 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'école de batterie partout dans le monde. La musique assistée par ordinateur (MAO), a elle aussi largement contribué à l'expansion des partitions de batterie puisque les compositeurs pouvaient utiliser des boîte à rythmes ou des séquenceurs pour écouter leurs productions et ainsi écrire une partition pour un batteur en s'émancipant de sa présence.

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'écritures. Reconnaître la métrique principale d'un rythme, la façon de regrouper les notes par les 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.

Nous proposons de rechercher des rythmes génériques (*motifs*) en amont dans la chaîne de traitement. Les *motifs* sont prédéfinis avec des combinaisons possibles (*gammes*) qui leur sont associées. Ces *motifs* et leur *gammes* respectives sont appelés *systèmes*. L'usage des *systèmes* a pour objectif de fixer des choix le plus tôt possible dans la chaîne de traitement afin de simplifier le reste des calculs en éliminant une partie d'entre eux. Ces choix concernent notamment la métrique, la séparation des voix ainsi

que les règles de réécriture.
Nous présenterons le contexte général suivi d'un état de l'art et nous définirons de manière générale le processus de transcription automatique de la musique pour enfin étayer les méthodes utilisées pour la transcription automatique de la batterie et nous présenterons les principales contributions apportées à l'outil qparse. Nous décrirons ensuite le corpus ainsi que les différentes expérimentations menées. Nous conclurons par une discussion sur les résultats obtenus et les pistes d'améliorations futures à explorer.

CHAPITRE 1

CONTEXTE

Sommaire

Introduction

L'ADT a engendré une pluie de sous-tâches qui ont donné naissance au MIR. Qu'est-ce que l'informatique musicale ? Quels sont les liens entre le MIR et le TAL ? Intérêt de l'ADT et problème de l'ADT ?
Digression sur la musicologie calculatoire (vs linguistique computationnelle).
Édition de partition.

Dans ce chapitre, nous présenterons le rapport possible entre la musique et le TAL, en considérant les notions de langage musical et langue naturelle, le lien entre partition musicale comme manière d'écrire la musique et texte comme manière d'écrire la parole.
Nous aborderons aussi les applications des techniques TAL pour les différents traitements associés à la musique et nous ferons un tour d'horizon des domaines de la recherche d'information musicale.
Nous présenterons ensuite le problème de la transcription automatique de la musique.

1.1 Informatique musicale

L'informatique musicale est une étude du traitement de la musique [1], en particulier des représentations musicales, de l'analyse de Fourier de la

musique, de la synchronisation de la musique, de l'analyse de la structure de la musique et de la reconnaissance des accords. D'autres sujets de recherche en informatique musicale comprennent la modélisation informatique de la musique (symbolique, distribuée, etc.), l'analyse informatique de la musique, la reconnaissance optique de la musique, les éditeurs audio numériques, les moteurs de recherche de musique en ligne, la recherche d'informations musicales et les questions cognitives dans la musique. [1]

1.2 TAL et MIR

Aborder la musique à travers le 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. Quelques travaux en neuroscience ont abordé la question, notamment par observation des processus cognitifs et neuronaux que les systèmes de traitement de ces deux langages avaient en communs. Dans le travail de Poulin-Charronnat et al. [2], 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. [3] 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 du sens d'une phrase musicale apparaît, selon eux, à la fois comme une limite et un défi majeur pour l'étude de cette analogie.
D'autre travaux très récents, ont aussi été révélé lors de la *première conférence sur le NLP pour la musique et l'audio (NLP4MusA 2020)*. Lors de cette conférence, Jiang et al. [4] ont présenté leur implémentation d'un modèle de langage musical auto-attentif 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.

1.3 La transcription automatique de la musique

Problème vieux et difficile ⇒ c'est un graal. Déjà en 1976, [5] évoquait la représentation musicale en arbre syntaxique.

1.3.1 Intérêt de l'ADT

Transcrire des solos, intérêt ⇒ constitution de corpus musicologique.
Voir l'intro de [6]

1. https://en.wikipedia.org/wiki/Music_informatics

L'objectif de la transcription automatique de la musique (AMT) [6] est de convertir la performance d'un musicien en notation musicale - un peu comme la conversion de la parole en texte dans le traitement du langage naturel. Bien que l'AMT soit un domaine de recherche en plein essor dans lequel plusieurs approches différentes sont encore activement étudiées, les performances des systèmes actuels ne sont pas encore suffisantes pour certaines applications qui exigent un haut degré de précision [6]. Même si les applications typiques de l'AMT comprennent l'estimation de la multi-tonalité, la classification des genres musicaux, la détection du début et de la fin des notes de musique, l'estimation du tempo, le suivi du rythme et la transcription de la musique. La plupart des travaux se sont concentrés sur le traitement du signal vers la génération du midi [7]. Seuls quelques travaux récents [8] s'intéressent de près à la création d'outils permettant la génération de partition. Le terme « transcription musicale automatique » a été utilisé pour la première fois par les chercheurs en audio James A. Moorer, Martin Piszczalski et Bernard Galler en 1977. Grâce à leurs connaissances en ingénierie audio numérique, ces chercheurs pensaient qu'un ordinateur pouvait être programmé pour analyser un enregistrement numérique de musique de manière à détecter les hauteurs des lignes mélodiques et des motifs d'accords, ainsi que les accents rythmiques des instruments à percussion.
La tâche de transcription automatique de la musique comprend deux activités distinctes : l'analyse d'un morceau de musique et l'impression d'une partition à partir de cette analyse. [2]

1.4 Les partitions

Mettre une image de partition ici Une partition de musique [3] est un document qui porte la représentation systématique du langage musical sous forme écrite. Cette représentation est appelée transcription et elle sert à traduire les quatre caractéristiques du son musical :

- la hauteur ;
- la durée ;
- l'intensité ;
- le timbre.

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).

2. https://en.wikipedia.org/wiki/Transcription_(music)
3. https://fr.wikipedia.org/wiki/Partition_(musique)

Exemple de sous-tâche dans la figure 1.1 remplace ARCHITECTURE

La figure suivante, qui est une proposition de Benetos et Al. [6], représente l'architecture générale d'un système de transcription musicale. *Les*

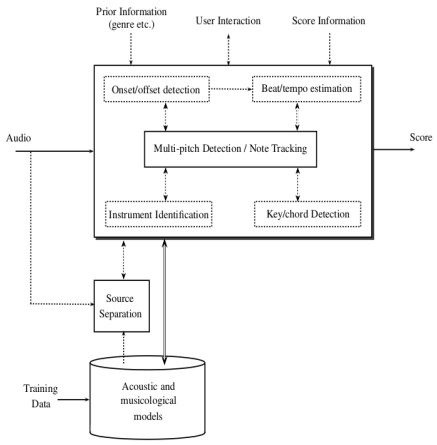


FIGURE 1.1 – Transcription automatique

sous-systèmes et algorithmes optionnels sont présentés à l'aide de lignes pointillées. Les doubles flèches mettent en évidence les connexions entre les systèmes qui incluent la fusion d'informations et une communication plus interactive entre les systèmes.

Au cœur du système se trouvent les algorithmes de détection des multi-pitchs et de suivi des notes. Quatre sous-tâches de transcription liées à la détection des hauteurs multiples et au suivi des notes apparaissent comme des algorithmes facultatifs du système (cases en pointillé) qui peuvent être intégrés dans un système de transcription. Il s'agit de l'identification de l'instrument, de l'estimation de la tonalité et de l'accord, de la détection de l'apparition et du décalage, et de l'estimation du tempo et du rythme. La séparation des sources, un problème indépendant mais lié, pourrait être traitée par un système séparé qui pourrait informer et interagir avec le système de transcription en général, et plus spécifiquement avec le sous-système d'identification des instruments. En option, des informations peuvent également être fournies de manière externe au système de transcription. Elles peuvent être données sous forme d'informations préalables (c'est-à-dire le genre, l'instrumentation, etc.), via l'interaction de l'utilisateur ou en fournissant des informations à partir d'une partition préexistante partiellement correcte ou incomplète. Enfin, les données de formation peuvent être utilisées pour apprendre des modèles acoustiques et musicologiques qui,

par la suite, informent le système de transcription et interagissent avec lui. avec le système de transcription.
Les applications de l'AMT ont aussi de la valeur dans les domaines oraux ou d'improvisation qui manquent de partition (jazz, pop) [6]. Les applications de l'ADT serait utile pour ces styles de musiques puisque la batterie y est amplement représentée. Un grand nombre travaux ont déjà été menés dans le domaine de l'ADT. La plupart ont été énumérés par Wu et al. [9] qui, pour mieux comprendre la pratique des systèmes d'ADT, se concentrent sur les méthodes basées sur la factorisation matricielle non négative et celles utilisant des réseaux neuronaux récurrents.
La batterie a un statuts à part dans l'univers de l'AMT puisqu'il s'agit d'instruments sans hauteur, d'événements auxquels une durée est rarement attribuée et de notations spécifiques (par exemple sur les têtes de notes). Si les ordinateurs étaient capables d'analyser la partie de la batterie dans la musique enregistrée, cela permettrait une variété de 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 [9]. Cependant, la plupart des travaux déjà entrepris se concentrent sur des méthodes de calcul pour la détection d'événements sonores de batterie à partir de signaux acoustiques ou sur la séparation entre les évènement sonore de batterie avec ceux des autres instruments dans un orchestre ou un groupe de musique [10], 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.

Conclusion

Dans le cas, de l'ADT, l'architecture reste la même mais de nombreuse seront à affiner, notamment pour les questions de continuation ainsi que celle des ghost-notes et des accents.

CHAPITRE 2

ÉTAT DE L'ART

Sommaire

Introduction

ainsi que les différentes avancées qui ont déjà eues lieues dans le domaine de la transcription de la musique. Et enfin, les avancées en terme de transcription automatique de la batterie. Actuellement, des modèles polyvalent qui n'arrivent pas à récupérer toute la richesse des sons sont utilisés.

2.1 Monophonique et Polyphonique

C'est le problème de la séparation des voix. Les premiers travaux ont été fait sur l'identification des instruments monophoniques (une seule note à la fois, ou plusieurs notes de même durée en cas de monophonie par accord). 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 polyphonique.
Le fort degrés de chevauchement entre les durées ainsi qu'entre les fréquences rendent l'identification des instruments polyphoniques difficile. Cette tâche est étroitement liées à la séparation des sources.
La création d'un système automatisé capable de transcrire de la musique polyphonique sans restrictions sur le degré de polyphonie ou le type d'ins-

trument reste encore ouverte. Un des principaux enjeux de ce problème est la détection des hauteurs de son multiples (F0 multiples) [6].

2.2 Audio vers MIDI

Voir : [6] - Multi-pitch détection and note tracking
- Détection of onsets and offsets - Instrument recognition
- Extraction of rhythmic information (tempo, beat, and musical timing)
- Estimation of pitch and harmony (key, chords and pitch spelling)

2.3 MIDI vers partition

2.3.1 Approche linéaire

nakamura [11]

2.3.2 Approche hiérarchique

[8] évoque la nécessité d'une approche hiérarchique pour la production automatique de partition même si la quantification du rythme se fait le plus souvent par la manipulation de données linéaires :
- rtu (real time units : secondes) vers mtu (musical time units : temps, métrique,...)
Dans [8], les modèles de grammaire exposés sont différents de modèles markoviens linéaires de précédent travaux. [12] [13]

Conclusion

Nous avons décidé de compléter le travail qui concerne la batterie en commençant par l'endroit le moins pratiqué, à savoir la transcription en partition pour à l'avenir réaliser la chaîne de bout en bout : de l'audio jusqu'à l'écriture de partition.

CHAPITRE 3

MÉTHODES ET CONTRIBUTIONS

Sommaire

3.1 Introduction

Dans ce chapitre, nous expliquerons en détails les méthodes que nous avons employées pour l'ADT. Nous commencerons par une description de qparse et des arbres de rythmes. Nous proposerons ensuite une modélisation comprenant une description de la notation de la batterie mise en relation avec les informations MIDI, ceci ayant pour objectif le parsing des données MIDI en arbre syntaxique. Enfin, nous démontrerons un modèle théorique de pattern (implémentable) qui devra être utilisé comme base de connaissance pour obtenir un système plus rapide et une meilleure qualité en sortie.

3.2 Les méthodes

3.2.1 Qparse

Qparse produit une partition musicale en prenant en entrée une performance musicale symbolique (par exemple un fichier MIDI) et un automate à arbre pondéré décrivant un langage de rythmes préférés (grammaire pondérée). La quantification des rythmes est basée sur des algorithmes d'analyse syntaxique applicables sur des automates arborescents. [1] En entrée : midi (séquence d'événements datés (piano roll) accompagné d'une grammaire pondérée)
⇒ parsing
⇒ global parsing tree
⇒ RI (Représentation Intermédiaire) arbres locaux par intruments
⇒ Sortie (xml, mei, lilypond,. . .)
Minimiser la distance entre le midi et la représentation en arbre.

3.2.2 Les données MIDI

MIDI (Musical Instrument Digital Interface) est une norme technique qui décrit un protocole de communication, une interface numérique et des connecteurs électriques permettant de connecter une grande variété d'instruments de musique électroniques, d'ordinateurs et d'appareils audio connexes pour jouer, éditer et enregistrer de la musique. [2]

Les données midi sont représentées sous forme de piano-roll. Chaque points sur la figure suivante est appelé « évènement midi » :

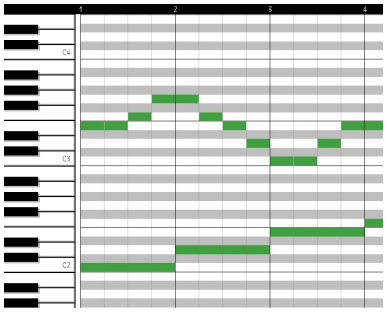


FIGURE 3.1 – Exemple évènements avec durée

1. https://qparse.gitlabpages.inria.fr
2. https://en.wikipedia.org/wiki/MIDI

Chaque évènement MIDI rassemble un ensemble d'informations sur la hauteur, la durée, le volume, etc. . . :

Protocol | Event

Property	Value
Type	Note On/Off Event
On Tick	15812
Off Tick	15905
Duration	93
Note	45
Velocity	89
Channel	9

FIGURE 3.2 – Critère pour un évènement

Pour la batterie, les évènements sont considérés sans durée, nous ignorerons donc les offsets (« Off Event »), les « Off Tick » et les « Duration ». Le channel ne nous sera pas utile non-plus. *Ici, définir Tick et channel.* Voici un exemple de piano-roll midi pour la batterie :

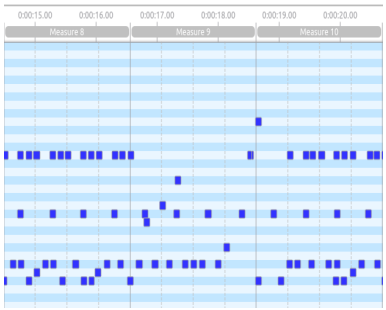


FIGURE 3.3 – Exemple évènements sans durée

On observe que toutes les durées sont identiques.

3.2.3 La grammaire pondérée

La grammaire pondérée qui accompagne le MIDI en input est une grammaire hors-contexte pondérée. Chaque règle comporte un poid qui sert à favoriser certains rythmes plutôt que d'autres. [3]

3.2.4 Les arbres de rythmes

Le parsing du midi donné en input crée une représentation symbolique sous forme d'arbre de rythme.
Ici ⇒ exemple avec :

3. https://qparse.gitlabpages.inria.fr/docs/scientific/

3bars_fill_groove-016.mid ⇒ arbre

3.2.5 La séparation des voix

Plusieurs écritures sont possibles pour un même rythme :

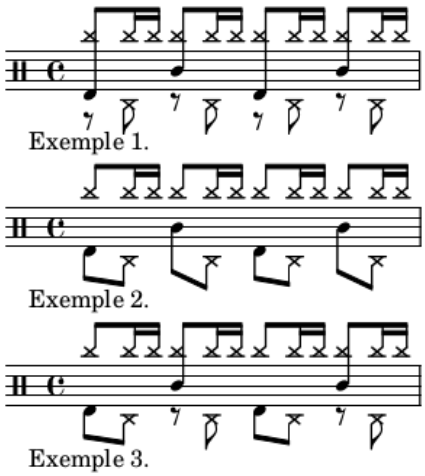


FIGURE 3.4 – Séparation des voix

Sur la figure 3.4, il faudra faire un choix entre les exemples 1, 2 et 3 qui sont trois façon d'écrire la même chose. Ce choix se fera en fonction de la lisibilité, de quelles instruments auront des phrasés plus ou moins chargé et/ou variés, auquel cas on les mettra dans une seule voix afin de ne pas charger la partition, etc. Ainsi l'arbre syntaxique de départ sera divisé en autant d'instruments qui le constituent et les voix seront regroupées de manière cohérentes.

3.2.6 Les règles de réécriture

Ici, description basique des règles de réécriture

3.3 Les contributions

3.3.1 La notation de la batterie

Les 3 parties d'une note en général :

- durée
- hampe
- tête de note (peut aussi indiquer la durée mais en batterie on évitera les blanches, etc.)

source : https://fr.wikipedia.org/wiki/Note_de_musique

Hauteurs et têtes de notes pour la batterie

Pour la transcriptions, nous proposons de choisir la base Agostini. La caisse claire centrale sur la portée est aussi centrale sur la batterie est elle est un élément qui conditionne la position des jambes (écart entre les pédales, etc.) ainsi que l'organisation des éléments en hauteur (toms, cymbales, etc.). On pensera en terme de symétrie la répartition des éléments par rapport au point central que constitue la caisse claire.
Cette symétrie s'opère en trois dimensions :

- Les hauteurs en terme de fréquences ;
- La hauteur physique des éléments :
  Du bas vers le haut : pédales, toms et caisse, cymbales
- L'ergonomie, qui hiérarchise l'importance des éléments sur la portée (caisse claire au centre, hh-pied et ride sont aux deux extrémités).

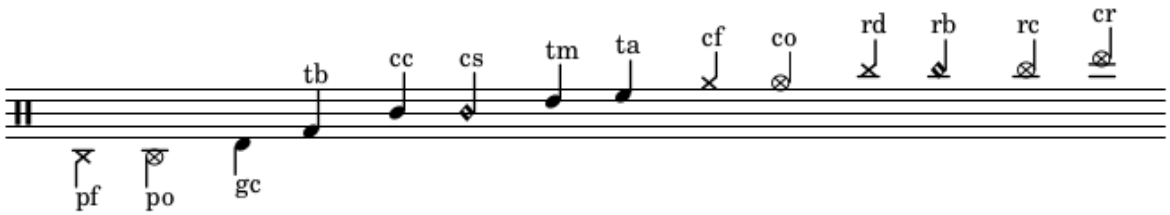


FIGURE 3.5 – Hauteur et têtes de notes

Les nuances

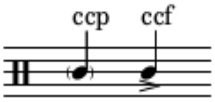


FIGURE 3.6 – Nuances

Bien expliquer les accents, remplacer p et f par g et a
⇒ nuance VS articulation

Les durées

Basé sur [14] et sur [15]
Pour la plupart des instruments mélodiques, la liaison et le point sont les deux seules possibilités en cas d'équivalence rythmique pour des notes dont la durée de l'une à l'autre est ininterrompue. Mais puisque les durées des notes n'ont pas d'importance en batterie, l'usage des silences

pour combler la distance rythmique entre deux notes devient possible.
Ceci pris en compte, et étant donné que les indications de durée dans les têtes de notes ne sont pas pratique en batterie (les symboles « x » des cymbales ne peuvent pas porter d'indication de durée dans la tête de notes [4]), l'écriture à l'aide de silences sera privilégiée comme indication de durée sauf dans les cas où cela reste impossible. Ce choix à pour but de n'avoir qu'une manière d'écrire toutes notes, que leurs têtes de notes soit modifiées ou non.

Exemple blanche vs noire + soupir

Les cymbales-crash et les ouvertures de charley constituent les seusl cas qui excluent cette option. Le charley car ses ouvertures/fermetures sont presque toujours quantifiées et les cymbales-crash car elles peuvent être arrêtées à la main de manière quantifié aussi mais ce cas est très rare, nous allons donc nous concentrer sur les ouvertures de charley et considérer les crashs comme des événements sans durée.
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.

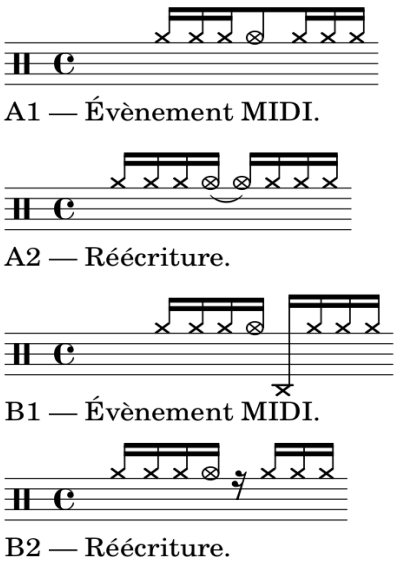


FIGURE 3.7 – Durées

4. Certains logiciels le permettent mais leur lecture reste peu aisée

3.3.2 Modélisation pour la transcription

Les pitchs

Codes	Instruments	Pitchs
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 3.1 – Pitchs et instruments

Pas de charley pied ouvert...

La vélocité

Codes	Instruments	Pitchs	Vélocité
cop	charley-main-ouvert	46	?

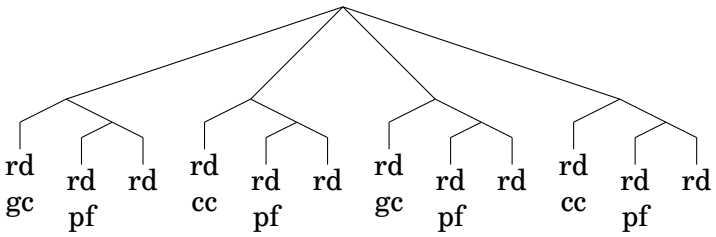
TABLE 3.2 – Vélocité et nuances

Nous ne prendrons en compte la vélocité que pour la cc, les toms et les cymbales jouées aux mains. Les nuances de grosse caisse et charley aux pieds sont le plus souvent insignifiantes, elles ne sont marquées sur le figure qu'à titre indicatif. Si la vélocité est en dessous de 40, il s'agit de ghost-notes : la tête de note devra être entouré de parenthèses et le suffixe *p (piano)* devra être ajouté au codes de l'instrument. (Voir ccp ci-dessus.) Si la vélocité est au dessus de 90, il s'agit de notes accentuées : le symbole « > » et le suffixe *f (forte)* devra être ajouté au codes de l'instrument. (Voir ccf ci-dessus.) Lorsque la vélocité va de 40 à 89, on considèrera le volume comme normal et aucun symbole supplémentaire ne sera ajouté à la note.

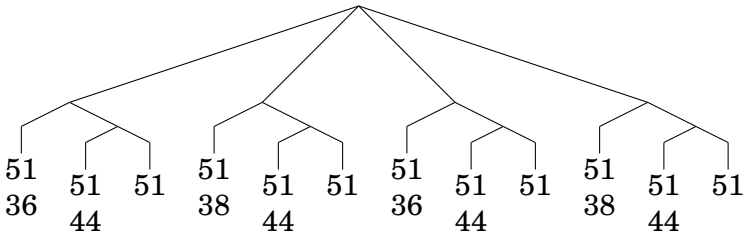
Le charley de pitch 46 est considéré comme le charley ouvert joué à la main sur le haut de la cymbale mais souvent, ça correspond au geste « tranche-olive » de la baguette lorsque le batteur accentue avec la tranche et joue moins fort avec l'olive sur le plat de la cymbale. Je vais dans un premier temps considérer le pitch comme **charley-main-ouvert-piano** (ghost-note)

Exemples de représentations en arbres

Voici une représentation de la *Figure 3.4* en arbre de rythme avec les codes de chaque instrument :



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



Cet arbre représente un rythme unique dont les possibilités de notation sur une partition sont théoriquement multiples. Les trois exemples de la figure 3.4 peuvent-être représentés par les arbres ci-dessus.

3.3.3 Les systèmes

Définition

Un système est la combinaison d'un ou plusieurs éléments qui jouent un rythme en boucle (motif) et d'un autre élément qui joue un texte rythmique variable mais respectant les règles propre au système (gamme).

Système = motif + gamme/texte
motif = rythmes coordonnés joués avec 2 ou 3 membres en boucle (reparti

sur 1 ou 2 voix)
gamme/texte = rythme irrégulier joué avec un seul membre sur le motif (Réparti sur 1 voix). La gamme d'un système considère l'ensemble des combinaisons que le batteur pourrait rencontrer en interprétant un texte rythmique à l'aide du système.

Nous partirons de propositions génériques de systèmes (environs trois systèmes dans différents styles de batterie) que nous tenterons de détecter dans le jeu de données groove.

Quatre systèmes standards :
- binaire
- ternaire (shuffle, afro, rock)
- jazz
- afro-cubain

Nous travaillerons aussi sur la détection de répétitions sur plusieurs mesures afin de pouvoir corriger des erreurs sur une des mesures qui aurait dû être identique aux autres mais qui présente des différences.

Intérêt des systèmes

Détection d'indication de mesure et choix de grammaire pondérée
Il faut prendre en compte l'existence potentielle de plusieurs grammaires *(un fichier wta par grammaire)* chacunes dédiées à un type de contenu MIDI. Le choix d'une grammaire pondérée doit être fait avant le parsing puisque qparse prends en entrée un fichier MIDI et un fichier wta.
Pour les expériences effectuées avec le Groove MIDI Data Set, le style et l'indication de mesure sont récupérables par les noms de fichiers MIDI, mais il faudra par la suite les trouver automatiquement sans autres indications que les données MIDI elles-mêmes.
En conséquence, les motifs des systèmes devront être recherchés sur l'input *(fichiers MIDI)* avant le lancement du parsing, afin de déterminer la métrique en amont en vu de sélectionner la grammaire pondérée *(fichier wta)* adéquate pour le parsing. Nous pensons que cette tâche devrait être effectuée en Machine Learning. **Les systèmes devront être matchés sur l'input MIDI**
- Définir une métrique ;
- Réécriture : séparation des voix ;
- Réécriture : Set de règles spécifiques de simplification.

Il faudra créer un ensemble de systèmes comprenant leurs règles spécifiques de réécriture (séparation des voix et simplifications). 3 grandes catégories :

Systèmes	Métriques	Subdivisions	Possibles	nb voix
binaires	simple	doubles-croches	triolets, sextolets	2
jazz	simple	triolets	croches et doubles-croches	2
ternaires	complexe	croches	duolets, quartelets	2
afros-cubains	simple	croches	-	3

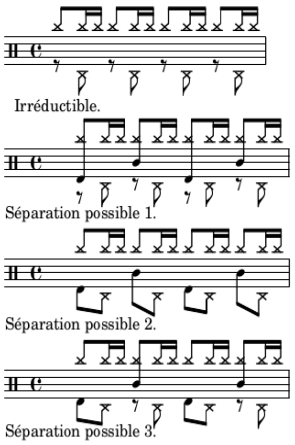
TABLE 3.3 – Sytèmes

— ternaire (mesures complexes, principalement croches, noire pointées, duolets et quartelets possibles)
— afro-cubain (mesure)
— Tout transcrire avec lilypond et en arbres d'analyse syntaxique.
— Créer les arbres de voix séparées.
— Écrire les règles de réécriture.
— Créer les arbres de voix séparées simplifiés (rewriting).

Pour la **séparation des voix** et la **définition des métriques**, nous nous intéresserons principalement à la partie *motif* des systèmes qui seront présentés. La partie *texte* nous intéressera plus pour les **combinaisons de réécritures**.

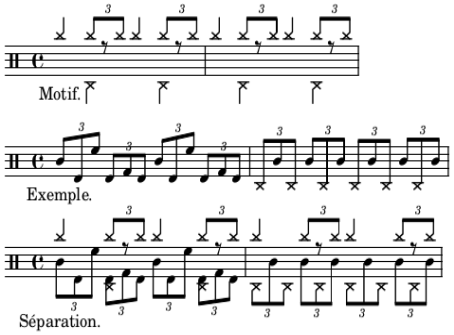
Réécriture — Pour la séparation des voix

Motif 4-4 binaire



Ici, le système est construit sur un modèle rock en 4/4 : after-beat sur les 2 et 4 avec un choix de répartition des cymbales type fast-jazz. Le système est constitué par défaut du motif ride/ch-pf/cc et d'un texte joué à la grosse-caisse. La troisième séparation proposée est privilégiée car elle répartit selon 2 voix, une voix pour les mains (ride + cc) et une voix pour les pieds (ch-pf + gc). Ce choix paraît plus équilibré car deux instruments sont utilisés par voix et plus logique pour le lecteur puisque les mains sont en haut et les pieds en bas.

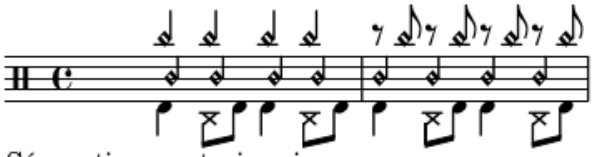
Motif 4-4 jazz



Dans la plupart des méthodes, le charley n'est pas écrit car considéré comme évident en jazz traditionnel. Ce qui facilite grandement l'écriture : la ride et les crash sur la voix du haut et le reste sur la voix du bas. Ici, le partie prit et de tout écrire. Dans l'exemple ci-dessus, les mesures 1 et 2 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 en haut et toms, caisse-claire, grosse-caisse et pédale de charley en bas.

Système 4-4 afro-cubain



Séparation sur trois voix.

Pour la reconnaissance de la métrique

12/8 vs 4/4 ternaire

Motif 12/8

Pour les règles de réécriture

Les textes qui accompagnent les motifs étayent toutes les combinaisons d'un systèmes.

Exemples à écrire en arbre :

— SI (pas pf) ET (note sur un temps suivie de note en l'air) :
⇒ (Temps1 : Note pertinente) + (Temps2 : Silence pertinent + Note pertinente.)

— Si (po ou co) déborde sur le temps suivant :
⇒ Liaison car marchera dans tous les cas même la où le point ne marchera pas (voir A2).

— Une blanche sera écrite noir + soupir.

3.3.4 Démonstration d'un modèle théorique

Partition de référence pour l'ouput

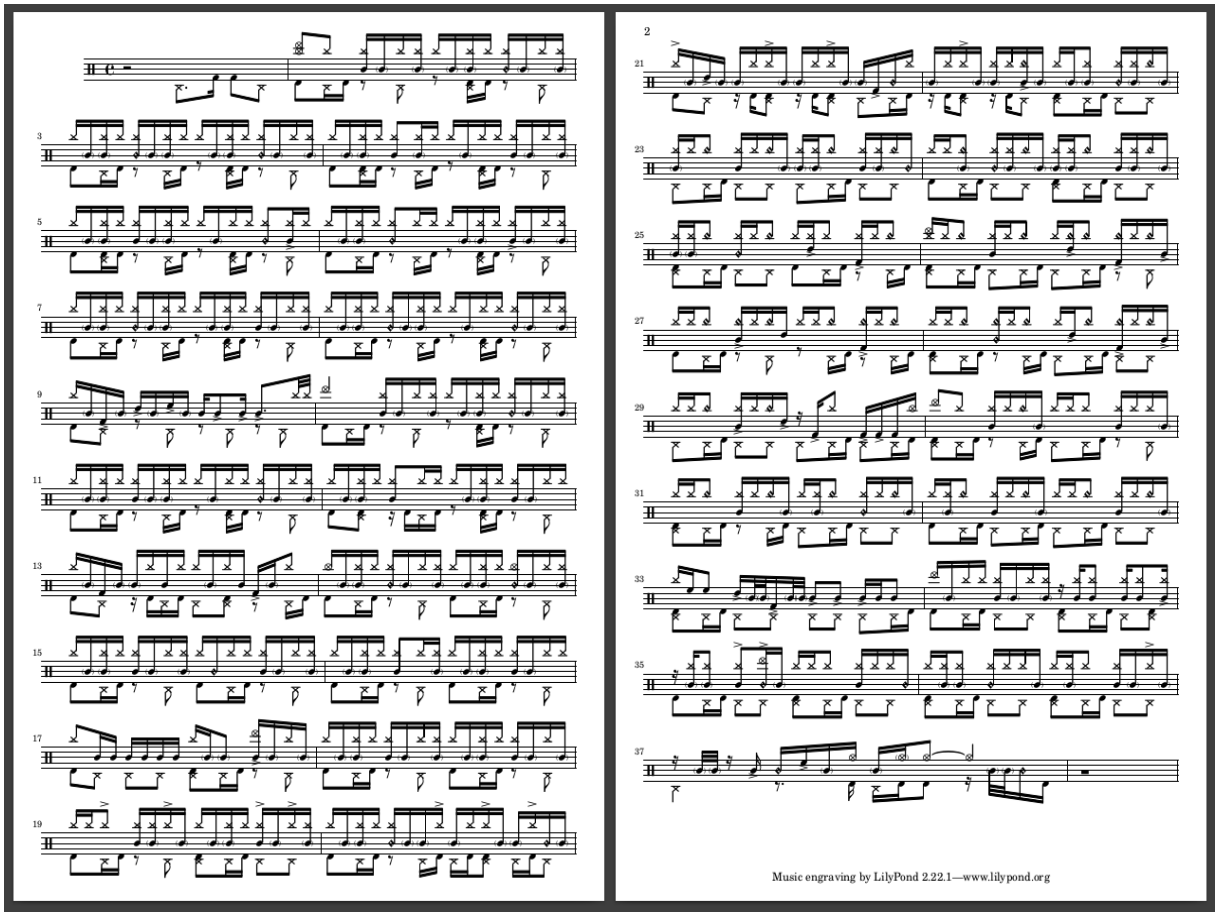


FIGURE 3.8 – référence 1

Il s'agit d'une partition d'un 4/4 binaire dont le fichier MIDI est annoncé dans le groove-dataset 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).

Systèmes recherchés

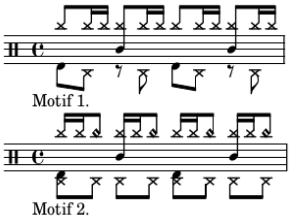
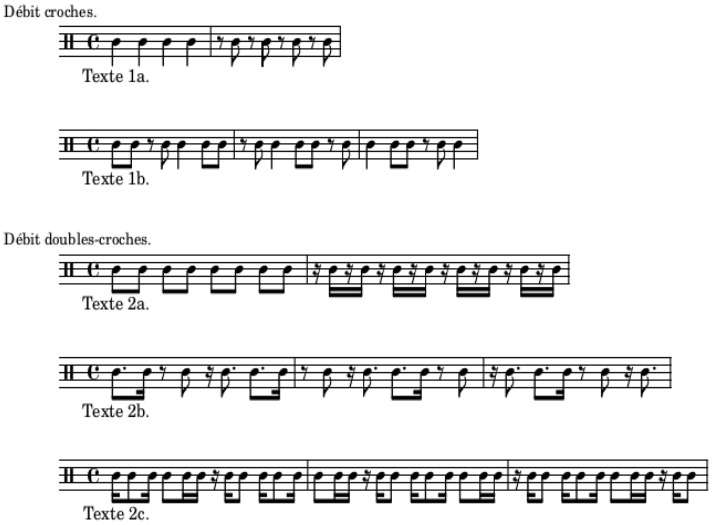


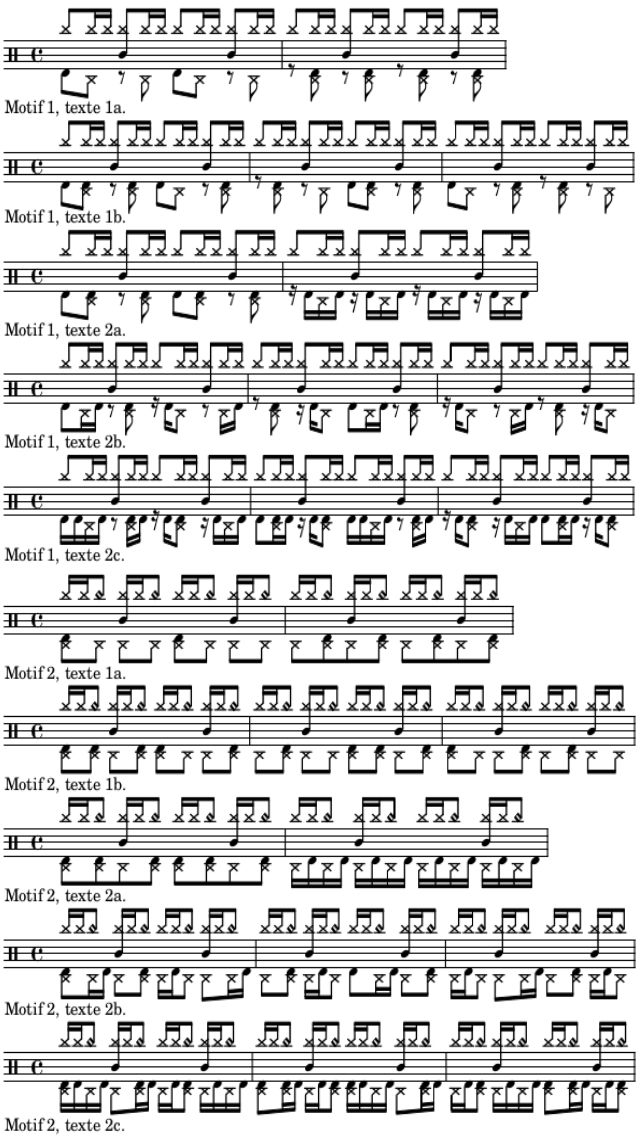
FIGURE 3.9 – Motifs

Les motifs 1 et 2 peuvent être extraits de la figure 3.8. Ces deux motifs sont très classiques et seront réutilisables aussi dans d'autres contextes. Le motif 1 est joué jusqu'à la mesure 18 avec des variations et des breaks. Le motif 2 est joué des mesures 23 à 28.

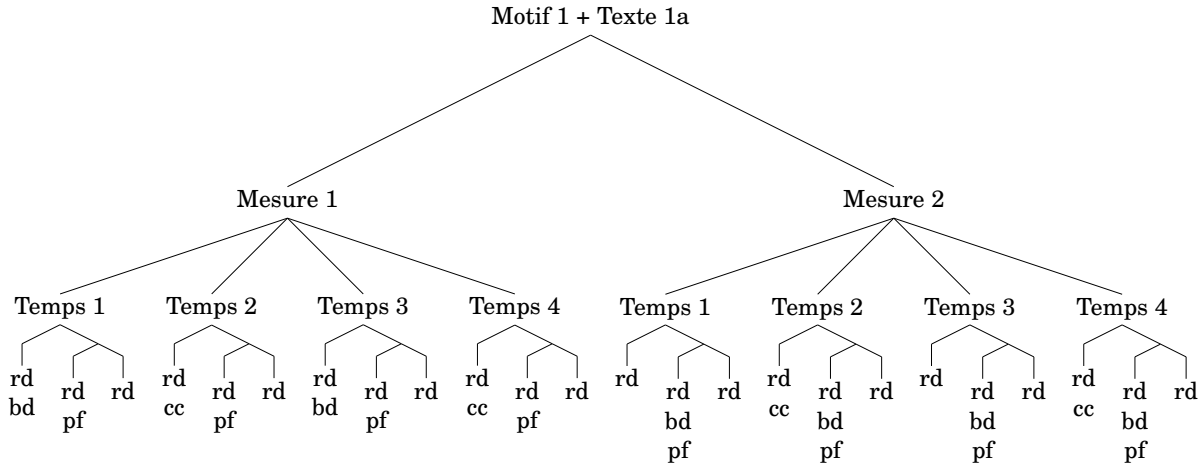
Gammes :



Systèmes résultants :

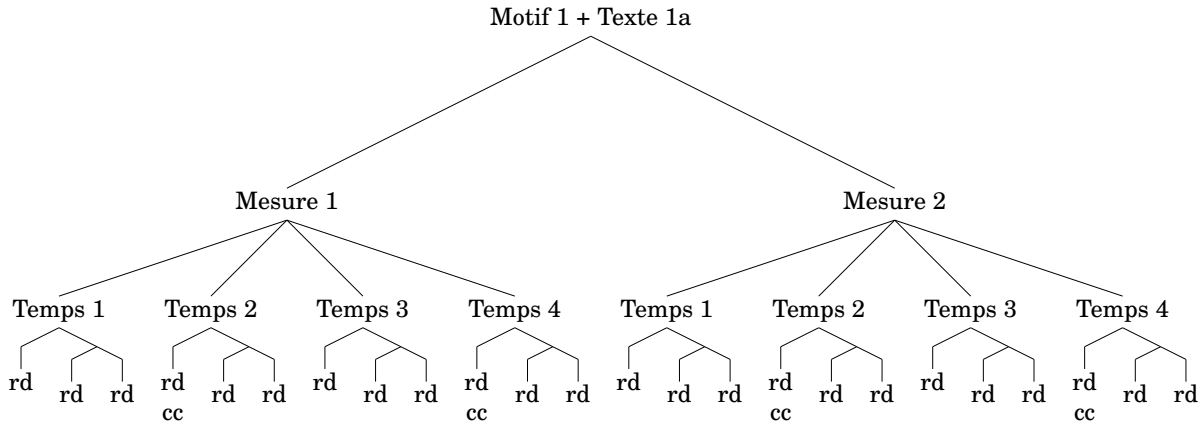

Motif 1, texte 1a.
Motif 1, texte 1b.
Motif 1, texte 2a.
Motif 1, texte 2b.
Motif 1, texte 2c.
Motif 2, texte 1a.
Motif 2, texte 1b.
Motif 2, texte 2a.
Motif 2, texte 2b.
Motif 2, texte 2c.

Représentation des systèmes en arbres de rythmes

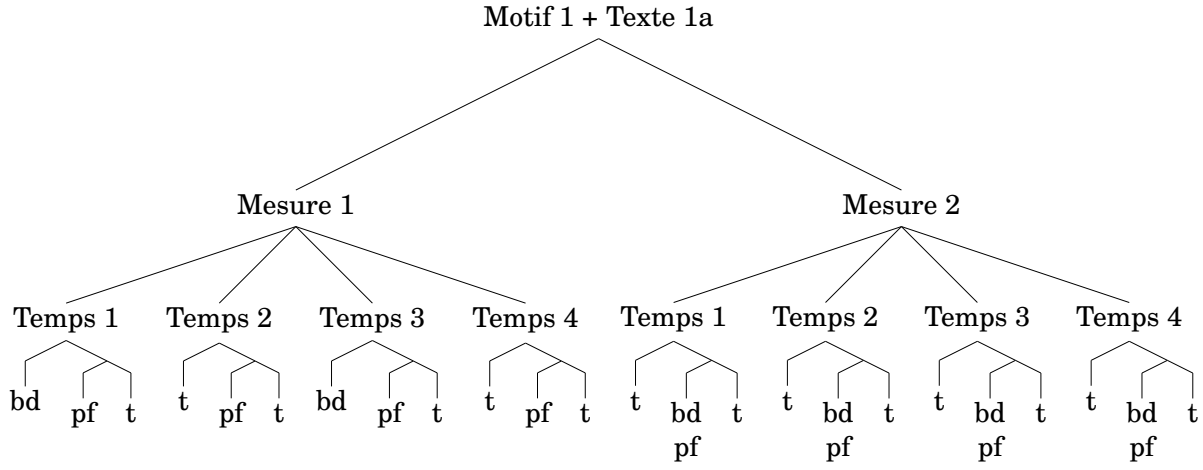


Séparation des voix

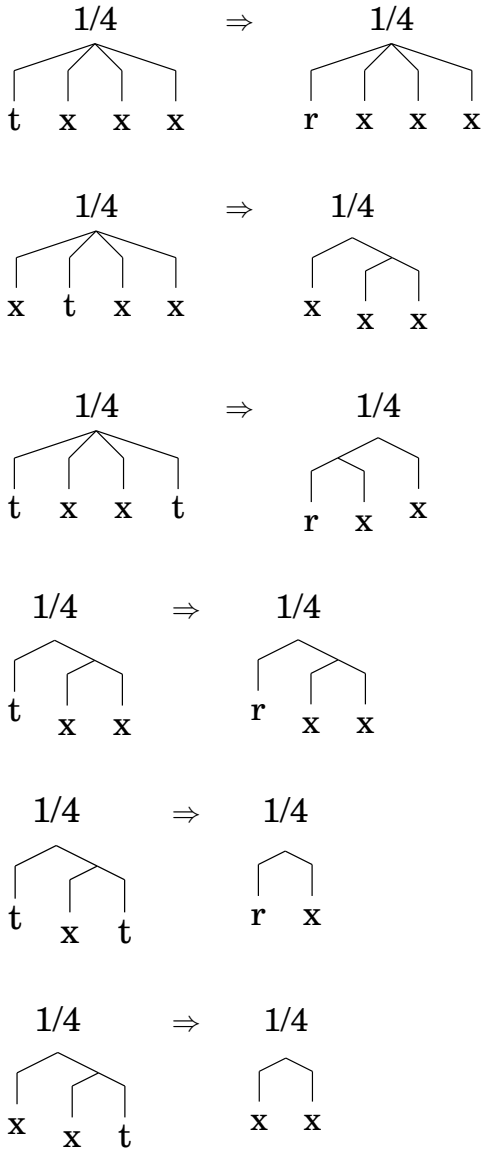
Voix haute



Voix basse



Règles de réécriture pour le 4/4 binaire



3.4 Conclusion

Bilan sur les différentes méthodes employées et la contribution que cela représente.

CHAPITRE 4

EXPÉRIMENTATIONS ET RÉSULTATS

Sommaire

4.1 Introduction

Dans ce chapitre, nous présenterons le corpus, les expérimentations et les différents choix effectués pour les tests.

4.2 Corpus et expérimentations

Le corpus

groove MIDI dataset
https://magenta.tensorflow.org/datasets/groove

- New sensing technology with Behavior Modeling provides a complete and natural drumming experience
- COACH and QUICK REC functions help build skills effectively at home, school, or any place of practice
- Large-sized LCD and simple user interface for easy operation
- USB-MIDI/AUDIO can conveniently connect to computer applications

Des batteurs pro ont été engagés pour jouer sur un roland td-11

The Groove MIDI Dataset (GMD), has several attributes that distinguish it from existing ones:

- The dataset contains about 13.6 hours, 1,150 MIDI files, and over 22,000 measures of drumming.
- Each performance was played along with a metronome set at a specific tempo by the drummer.
- The data includes performances by a total of 10 drummers, with more than 80% of duration coming from hired professionals. The professionals were able to improvise in a wide range of styles, resulting in a diverse dataset.
- The drummers were instructed to play a mix of long sequences (several minutes of continuous playing) and short beats and fills.
- Each performance is annotated with a genre (provided by the drummer), tempo, and anonymized drummer ID.
- Most of the performances are in 4/4 time, with a few examples from other time signatures.
- Four drummers were asked to record the same set of 10 beats in their own style. These are included in the test set split, labeled `eval-session/groove1-10`.
- In addition to the MIDI recordings that are the primary source of data for the experiments in this work, we captured the synthesized audio outputs of the drum set and aligned them to within 2ms of the corresponding MIDI files.

Les métadatas :

The metadata file (`info.csv`) has the following fields for every MIDI/WAV pair:

Field	Description
drummer	An anonymous string ID for the drummer of the performance.
session	A string ID for the recording session (unique per drummer).
id	A unique string ID for the performance.
style	A string style for the performance formatted as "<primary>/<secondary>". The primary style comes from the Genre List below.
bpm	An integer tempo in beats per minute for the performance.
beat_type	Either "beat" or "fill"
time_signature	The time signature for the performance formatted as "<numerator>-<denominator>".
midi_filename	Relative path to the MIDI file.
audio_filename	Relative path to the WAV file (if present).
duration	The float duration in seconds (of the MIDI).
split	The predefined split the performance is a part of. One of "train", "validation", or "test".

Genre List: afrobeat, afrocuban, blues, country, dance, funk, gospel, highlife, hiphop, jazz, latin, middleeastern, neworleans, pop, punk, reggae, rock, soul

A train/validation/test split configuration is provided for easier comparison of model accuracy on various tasks.

Split	Beats	Fills	Measures (approx.)	Hits	Duration (minutes)
Train	378	519	17752	357618	648.5
Validation	48	76	2269	44044	82.2
Test	77	52	2193	43832	84.3
Total	**503**	**647**	**22214**	**445494**	**815.0**

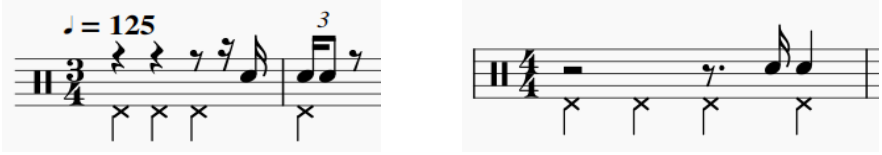
Détails (entre autres tensorflow avec le dataset) à : https://magenta.tensorflow.org/datasets/groove#license
écouter le dataset groove

Les expérimentations

Comparaisons de transcriptions

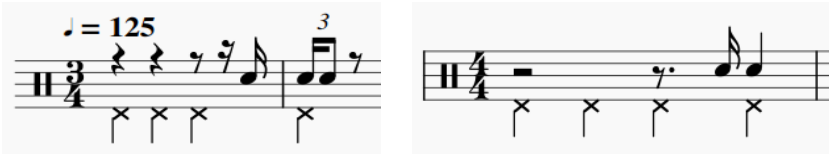
drummer_01/session3 — 10_rock-folk_90_beat_4-4

Fichier midi vers partition avec musescore ⇒ Transcription manuelle



drummer_01/session3 — 10_rock-folk_90_beat_4-4

Fichier midi vers partition avec musescore ⇒ Transcription manuelle

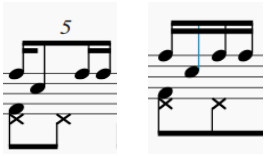


— Erreur d'indication de mesure ;
— Mauvaise transcription d'une noire.

La noire du 4ème temps se retrouve sur le premier temps de la mesure suivante et elle se transforme en un triolet de double croches dont seules les deux premières seraient jouées.

drummer_01/session3 — 10_rock-folk_90_beat_4-4

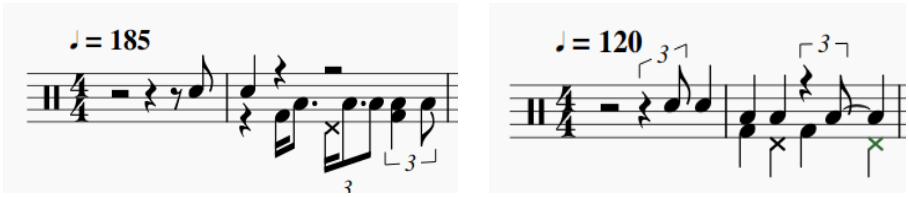
Fichier midi vers partition avec musescore ⇒ Transcription manuelle



— Erreur de quantification : les doubles croches ont été interprétées en quintolet ;

drummer_01/session3 — 2_jazz-swing_185_beat_4-4

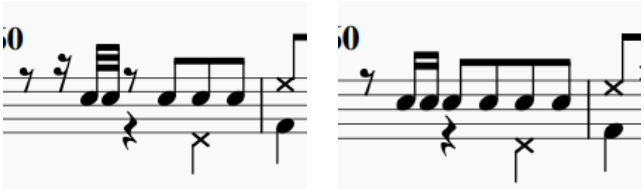
Fichier midi vers partition avec musescore ⇒ Transcription manuelle



— L'indication de mesure est correcte mais tout a été décalé d'un temps car la première noire sur la caisse claire est jouée sur le 4ème temps et non sur le premier temps de la deuxième mesure comme l'indique la transcription de musescore.
— Les toms basses des 1er et 2ème temps de la mesure musescore auraient dû être sur les temps et non décalés d'une double croche vers la droite.

drummer_01/session1 — 1_funk_80_beat_4-4

Fichier midi vers partition avec musescore ⇒ Transcription manuelle



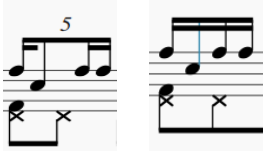
— On dirait que lorsque certaines notes sont proches, elles se resserrent et suppriment celles qui aurait dû être sur le temps.

— Erreur d'indication de mesure ;
— Mauvaise transcription d'une noire.

La noire du 4ème temps se retrouve sur le premier temps de la mesure suivante et elle se transforme en un triolet de double croches dont seules les deux premières seraient jouées.

drummer_01/session3 — 10_rock-folk_90_beat_4-4

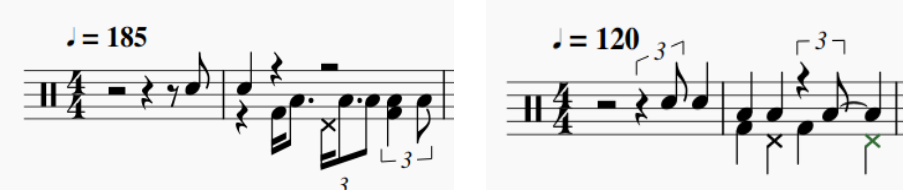
Fichier midi vers partition avec musescore ⇒ Transcription manuelle



— Erreur de quantification : les doubles croches ont été interprétées en quintolet ;

drummer_01/session3 — 2_jazz-swing_185_beat_4-4

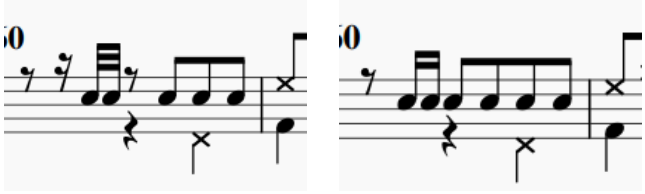
Fichier midi vers partition avec musescore ⇒ Transcription manuelle



— L'indication de mesure est correcte mais tout a été décalé d'un temps car la première noire sur la caisse claire est jouée sur le 4ème temps et non sur le premier temps de la deuxième mesure comme l'indique la transcription de musescore.
— Les toms basses des 1er et 2ème temps de la mesure musescore auraient dû être sur les temps et non décalés d'une double croche vers la droite.

drummer_01/session1 — 1_funk_80_beat_4-4

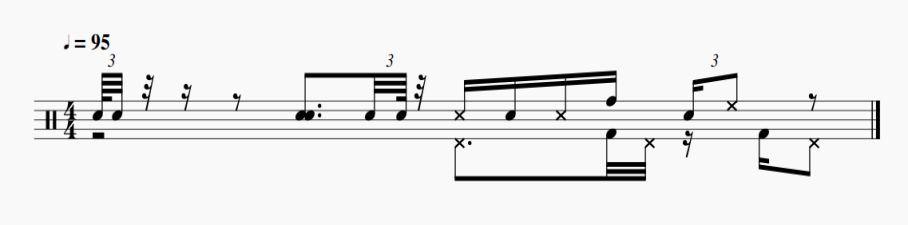
Fichier midi vers partition avec musescore ⇒ Transcription manuelle



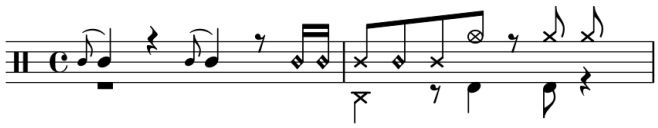
— On dirait que lorsque certaines notes sont proches, elles se resserrent et suppriment celles qui aurait dû être sur le temps.

Exemple avec des flas

Fichier midi vers partition avec musescore :



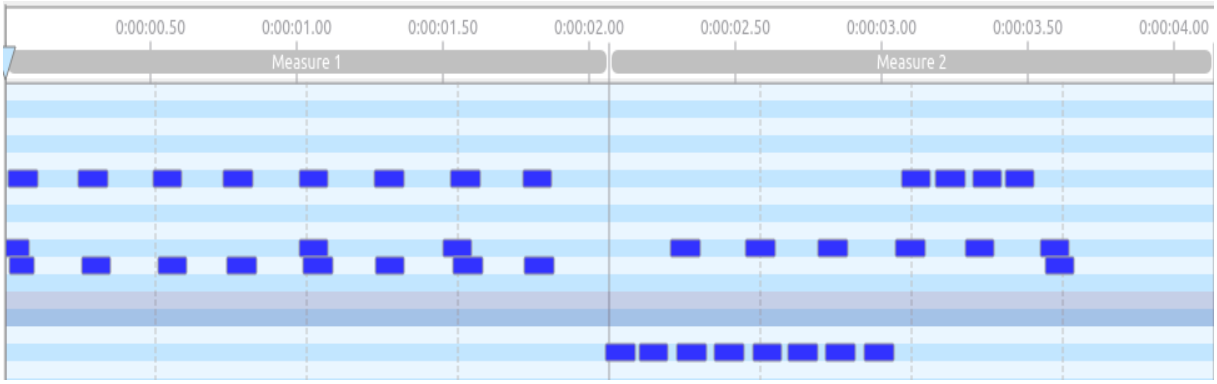
Transcription manuelle :



Les tests unitaires

Le parsing avec squant

Tests effectués avec le fichier midi suivant :



Un premier test convaincant est effectué avec la grammaire suivante :

// bar level
0 -> C0 1
0 -> E1 1
0 -> U4(1, 1, 1, 1) 1

// half bar level
9 -> C0 1
9 -> E1 1

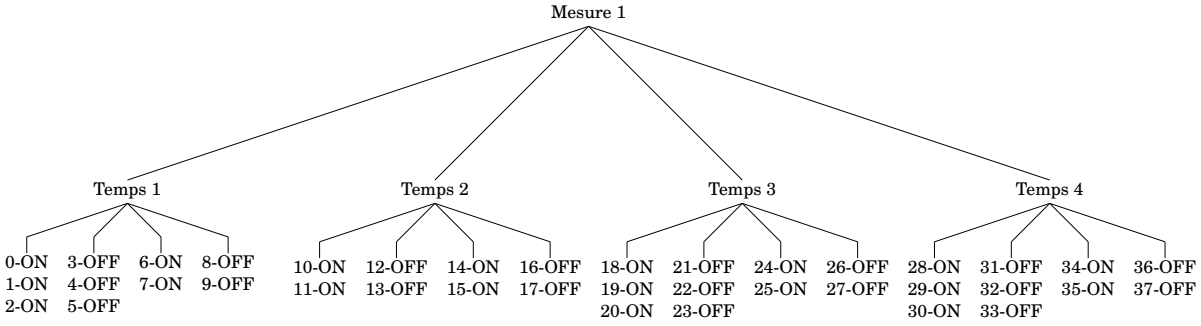
// beat level
1 -> C0 1
1 -> E1 1
1 -> T2(2, 2) 1
1 -> T4(4, 4, 4, 4) 1

// croche level
2 -> C0 1
2 -> E1 1

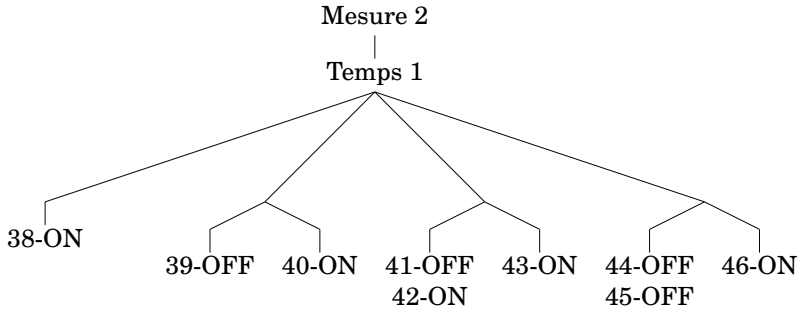
// double level
4 -> C0 1
4 -> E1 1
4 -> E2 1
4 -> T2(6, 6) 1

// triple level
6 -> E1 1

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



Les temps de la première mesure du fichier MIDI sont bien quantifié mais ceux de la deuxième mesure présentent quelques défauts de quantification visibles dès le premier temps.



Les Onsets sont correctement triés au niveau des doubles croches mais certaines doubles croches sont inutilement subdivisées en triples croches (les 2ème, 3ème et 4ème doubles croches sur le premier temps ci-dessus).

2ème tentative :

Après une augmentation du poids des triples croches dans la grammaire (monté de 1 à 5)et une baisse de tous les autres poids (descendu de 1 à 0.5), et mis à part le troisième temps de la 2ème mesure, tous les Onsets sont bien triés et aucuns ne sont subdivisés.

4.2.1 Reconnaissance d'un motif sur l'arbre de parsing

Reconnaître un motif (système) sur une mesure de l'input (un fichier midi représentant des données audios)
⇒ Motif (système) reconnu : true ou false
Si true, appliquer les règles de réécritures.

4.2.2 Réécriture

Séparation des voix

Règles établies par le système

Règles de simplifications

Simplifier l'écriture de chaque voix (*Règles établis par le système*)
Contribution sur la branch « distance » dans :

— qparselib/notes/cluster.md
— qparselib/src/segment/import/ :
DrumCode hpp et cpp

4.3 Résultats et discussion

4.3.1 Résultats

Essayer d'implémenter un pattern...

4.3.2 Évaluation

1 - Transcription manuelle à partir de fichier midi et/ou wav d'une partition contenant des systèmes. Écriture des systèmes contenues dans la partition (arbres, séparation des voix, réécriture)

2 - Trouver un moyen de comparer l'arbre obtenu automatiquement de l'arbre de la transcription manuelle.

4.4 Discussion

4.4.1 Machine learning

4.4.2 Travaux futurs

Le ternaire jazz (voir expérience 2)

4.5 Conclusion

Conclusion de ce chapitre.

CONCLUSION GÉNÉRALE

Conclusion : la conclusion globale du mémoire.

Dans ce mémoire, nous avons traité de la problématique...

L'intégration de ces LM avec l'état de l'art des modèles acoustiques (AM) et des méthodes pour les tâches de traitement du signal ci-dessus. Cela nécessite la prise en compte du contexte musical et des informations musicales de haut niveau des LM en plus des caractéristiques acoustiques de bas niveau ci-dessus.
En outre, certaines expériences seront menées sur la base d'ensembles de données publiques, afin d'évaluer l'approche intégrée. Elles devraient couvrir certains cas de motifs rythmiques complexes se chevauchant.

Au-delà de l'intégration de modèles, 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. [9]

BIBLIOGRAPHIE

[1] Meinard Müller. *Fundamentals of Music Processing*. 01 2015. – Cité page 7.

[2] 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 8.

[3] 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 8.

[4] Junyan Jiang, Gus Xia, and Taylor Berg-Kirkpatrick. Discovering music relations with sequential attention. In *NLP4MUSA*, 2020. – Cité page 8.

[5] H. C. Longuet-Higgins. Perception of melodies. 1976. – Cité page 8.

[6] Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Holger Kirchhoff, and Anssi Klapuri. Automatic music transcription : Challenges and future directions. *Journal of Intelligent Information Systems*, 41, 12 2013. – Cité pages 8, 9, 10, 11 et 14.

[7] Moshekwa Malatji. Automatic music transcription for two instruments based variable q-transform and deep learning methods, 10 2020. – Cité page 9.

[8] 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 9 et 14.

[9] 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 11 et 43.

[10] 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 11.

[11] 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 page 14.

[12] Daniel Harasim, Christoph Finkensiep, Petter Ericson, Timothy J O'Donnell, and Martin Rohrmeier. The jazz harmony treebank. – Cité page 14.

[13] Martin Rohrmeier. Towards a formalisation of musical rhythm. In *Proceedings of the 21st Int. Society for Music Information Retrieval Conf*, 2020. – Cité page 14.

[14] 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 19.

[15] 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)*, Coroña, Spain, May 2017. Helena Lopez Palma and Mike Solomon. – Cité page 19.