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

QUOI ?
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 ?
→ Problèmatique :
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 métrique 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 3.4, 4.4). 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 métrique 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 3.2) qui était jusqu'à présent inexistante.
- plusieurs trancriptions manuelles dans le but d'analyser les contenus des fichiers MIDI et Audio (4.2) et de faire des comparaisons de transcription avec des outils déjà existants [1].
- 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 (4.2). 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 [2] et l'implémentation de tests unitaires sur le traitement des Jams (4.3).
- la création de grammaires pondérées spécifiques à la batterie (4.3)

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 4.4 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 conclurons 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.

CHAPITRE 1

CONTEXTE

Sommaire

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) [1]. 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 [2]. <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 [3], 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 [4].

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 [5] [6] [7] 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].

exemples / illustration de la proximité thématique ?

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	signal ⇒ phonèmes ⇒ texte	données linéaires
AMT	signal ⇒ notes, accords ⇒ partition	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 [8] (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 [9], ...). La TAM est aussi utile pour la recherche et l'annotation automatique d'informations musicales, pour l'analyse musicologique [10] 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 [11].

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 [12] ou de séquenceurs [13] 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.

cite méthode et école Agostini ?

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

citer M. Müller FMP pour cette section ?

Les données audio

trop technique. ne pas recopier wikipédia

Le format de fichier WAV est une instance du *Resource Interchange File Format (RIFF)* défini par IBM et Microsoft. Le format RIFF agit comme une "enveloppe" pour divers formats de codage audio. Un fichier WAV peux contenir de l'audio compressé ou non compressé.

LPCM pas utile ici. parle juste échantillons et compression.

Les données MIDI

tu peux mentionner le format spectral (analyse harmonique) crucial en MIR audio.

Le MIDI [14] (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.

ne pas copier wikipédia vert batim. source : midi.org MIDI est un protocole temps réel pour échanger des messages (événement) et un format de fichier.

Les données midi sont représentées sous forme de piano-roll. Chaque point sur la figure 1.1 est appelé « évènement MIDI » :

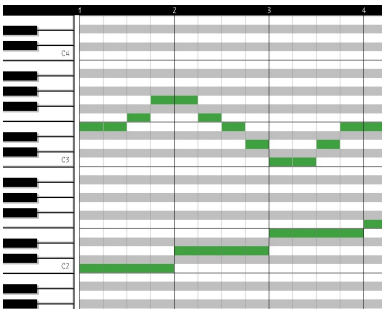


FIGURE 1.1 – Exemple évènements avec durée

fichier MIDI = séquence événements MIDI + dates (timestamp) performance musicale symbolique

donner ici les données des événements et expliquer ON/OFF (clavier)

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

il n'y a pas de duration d'événement dans un MIDI file. la "durée" est une distance entre 2 événemtns ON et OFF (c'est important dans ton travail). le screen shot n'est pas utile, écrit plutôt une liste itemize

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 :

14. https://en.wikipedia.org/wiki/MIDI

Protocol | Event

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

FIGURE 1.2 – Critère pour un évènement

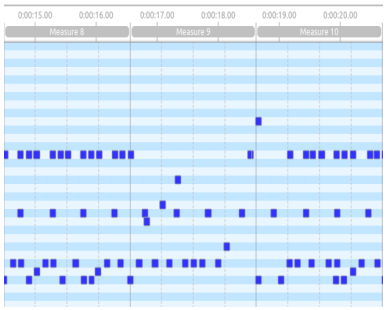


FIGURE 1.3 – Exemple évènements sans durée

On observe que toutes les durées sont identiques. <dam>je te suggère un petit paragraphe ensuite, genre : "Le format MIDI, originellement une norme technique, peut également être considéré comme une représentation musicale. Celle-ci peut effectivement être visualisée sous la forme d'une partition ou jouée par 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)"</dam>

Les partitions

Une partition de musique [15] 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 ;

15. https://fr.wikipedia.org/wiki/Partition_(musique)

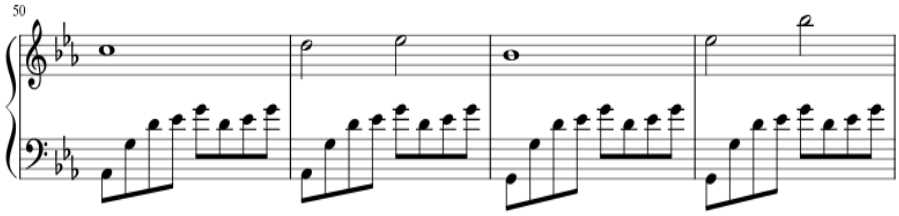


FIGURE 1.4 – Exemple de partition de piano

expliquer un peu plus avec exemple. ce serait mieux d'avoir un ex. avec des nuances, accents, appogiatures...

— 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).

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

Les formats XML

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.

```
<?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>
```

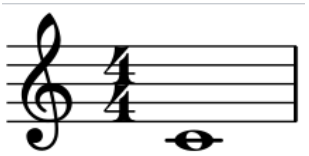


FIGURE 1.5 – MusicXML

Le figure 1.5 [16] 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 une interface homme/machine.

16. *Source images :* https://fr.wikipedia.org/wiki/MusicXML

Conclusion

Dans ce chapitre, nous avons établi que la RIM s'intéresse de plus en plus au TAL, et que, par ce biais, il y a des liens possibles entre le langage musical et les langues naturelles, le plus proche étant probablement le phénomène d'écriture des sons de l'un comme de l'autre.
Nous avons également établi que la RIM est née de la TAM qui est un problème ancien et très difficile et qu'il serait toujours très utile de le résoudre (autant pour la TAM que pour la TAB).
Et enfin, nous avons décrit les représentations de la musique nécessaires à la compréhension du présent mémoire, allant du son jusqu'à l'écriture.

CHAPITRE 2

ÉTAT DE L'ART

Sommaire

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 [1] [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 [2]. L'estimation des hauteurs multiples

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. . .

(détection multi-pitchs ou F0 multiples) est le problème central de la création d'un système de transcription de musique polyphonique. Il s'agit de la détection de notes qui peuvent apparaître simultanément et être produites par plusieurs instruments différents. Ce défi est donc majeur pour la batterie puisque c'est un instrument qui est lui-même constitué de plusieurs instruments (caisse-claire, grosse-caisse, cymbales, toms, etc...). Le fort degré de chevauchement entre les durées ainsi qu'entre les fréquences complique l'identification des instruments polyphoniques. Cette tâche est étroitement liée à la séparation des sources et concerne aussi la séparation des voix. 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 Audio vers MIDI

Jusqu'à aujourd'hui, les recherches se sont majoritairement concentrées sur le traitement de signaux audio vers la génération du MIDI [15].
Cette partie englobe plusieurs sous-tâches dont la détection multi-pitchs, la détection des onset et des offset, l'estimation du tempo, la quantification du rythme, la classification des genres musicaux, etc...
La figure 2.1 est une proposition de Benetos *et al.* [11] qui représente l'architecture générale d'un système de transcription musicale. On y observe plusieurs sous-tâches de la TAM :

- La séparation des sources à partir de l'audio.
- Le système de transcription :
  - Cœur du système :
    ⇒ Algorithmes de détection des multi-pitchs<dam>un autre terme plus compréhensible ?</dam> et de suivi des notes.
    Quatres sous-tâches optionnelles accompagnent ces algorithmes :
    - identification de l'instrument ;
    - estimation de la tonalité et de l'accord ;
    - détection de l'apparition et du décalage ;
    - estimation du tempo et du rythme.
    - <dam> ça serait bien d'avoir une vision approximative des données : - identification de l'instrument : valeur symbolique prise dans une liste prédéfinie ? - estimation de la tonalité et de l'accord : en note la gamme ou Hz ? - détection de l'apparition et du décalage : mesure de temps / durée - estimation du tempo et du rythme :?</dam>
- Apprentissage sur des modèles accoustiques et musicologiques.

MIDI **non-quantifié** = performance (à expliquer)

en général tempo et quantification ne sont pas traités ici, le but est seulement la génération d'un MIDI non quantifié

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.

classification des genres ? ce n'est pas de la transcription! séparation des sources oui.

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

la figure ne correspond pas à ton travail. ici "score" = MIDI performance.

— *Optionnel :* Informations fournies de manière externe, soit fournie en amont (genre, instruments,...), soit par interaction avec un utilisateur (infos sur une partition incomplète).

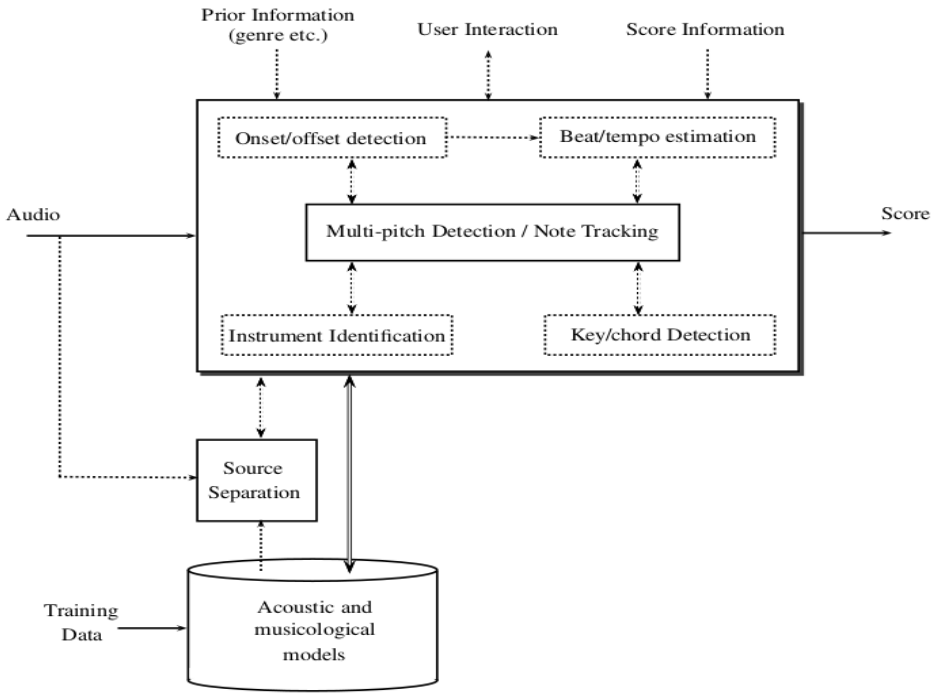


FIGURE 2.1 – Transcription automatique <dam>remettre ici la citation de la capture d'écran avec la page</dam>

Les 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.

En ADT [14], plusieurs stratégies de répartition pré/post-processing sont possibles pour la détection multi-pitchs. Entamer la détection dès le pré-processing, en supprimant les features non-pertinentes pendant la séparation des sources afin d'obtenir une meilleure détection des instruments de la batterie, est une démarche intuitive : supprimer la structure harmonique pour atténuer l'influence des instruments à hauteurs sur la détection grosse-caisse et caisse-claire en est un exemple. Mais certaines études montrent que des expériences similaires ont donné des résultats non-concluants et que la suppression des instruments à hauteurs peut avoir des effets néfastes sur les performances de l'ADT. En outre, les systèmes d'ADT basés sur des réseaux de neurones récurrents (RNN) ou sur des factorisations matricielles non négative font la séparation des sources pendant l'optimisation, ce qui réduit la nécessité de la faire pendant le

haute fréquence, aigus ?

pré-processing.
Pour la reconnaissance des instruments, une approche possible [16] est de mettre un modèle probabiliste dans l'étape de la classification des évènements afin de classer les différents sons de la batterie. Cette méthode permet de se passer de samples audio isolés en modélisant la progression temporelle des *features* [3] avec un modèle de markow caché (HMM). Les *features* sont transformés en représentations statistiques indépendantes.
L'approche AdaMa [17] est une autre approche de la même catégorie ; elle commence par une estimation initiale des sons de la batterie qui sont itérativement raffinés pour correspondre à (pour matcher) l'enregistrement visé.

classification des évènements ? la phrase semble redondante

pas clair... peut–être juste mentionner les modèles probabilistes utilisés

2.3 MIDI vers partition

Le plus souvent, lorsque les articles abordent la transcription automatique de bout en bout (de l'audio à la partition), l'appellation « *score* » (partition) désigne un ouput au format Music XML, ou simplement MIDI. Par exemple, dans [18], la chaîne de traitement va jusqu'à la génération d'une séquence MIDI quantifiée qui est importée dans MuseScore pour en extraire manuellement un fichier MusicXML contenant plusieurs voix.
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'évènements musicaux symboliques en une partition musicale structurée est traité notamment dans [19]. Ce travail, qui vise à résoudre en une fois la quantification rythmique et la production de partition structurée, s'appuie tout au long du processus sur des grammaires génératives qui fournissent un modèle hiérarchique *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, en couplant le problème de la séparation des voix avec la quantification du rythme.
L'approche de [19] est fondée sur la conviction que la complexité de la structure musicale dépasse les modèles linéaires.

ce n'est pas exactement cela. cf. proposition de description + détaillée en commentaires

de manière conjointe

langage a priori

qui nécessite de traiter le problème supplémentaire de la séparation de voix. i.e. pour la batterie on nveut quantification + structuration + séparation mais seules les 2 premières sont couplées dans l'approche de tonn stage.

2.4 Approche linéaire et approche hiérarchique

Plusieurs travaux ont d'abord privilégié l'approche stochastique. Par exemple, Shibata *et al.* [18] ont utilisé le modèle de Markov caché (HMM) [4] pour la reconnaissance de la métrique. Les auteurs utilisent d'abord deux

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

4. https://fr.wikipedia.org/wiki/Modèle_de_Markov_caché
https://en.wikipedia.org/wiki/Hidden_Markov_model

558 réseaux de neurones profonds, l'un pour la reconnaissance des pitchs et
559 l'autre pour la reconnaissance de la vélocité. Pour la dernière couche, la
560 probabilité est obtenue par une fonction sigmoïde. Ils construisent en-
561 suite plusieurs HMM métriques étendus pour la musique polyphonique
562 correspondant à des métriques possibles, puis ils calculent la probalitité
563 maximale pour chaque modèle afin d'obtenir la métrique la plus probable.

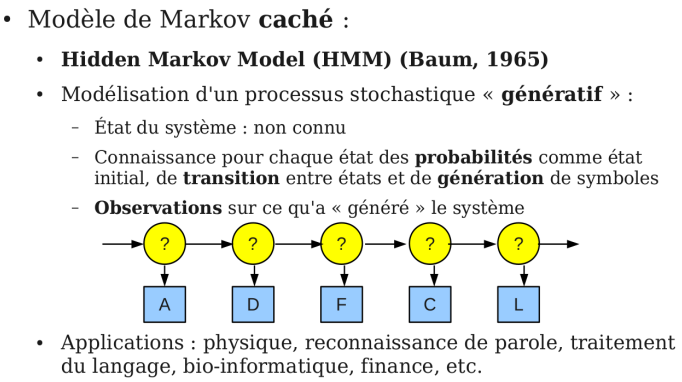


FIGURE 2.2 – HMM

Source : Cours de Damien Nouvel [5]

L'évaluation finale des résultats de [18] 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 l'exécution de la musique. Or, même si la quantification du rythme se fait le plus souvent par la manipulation de données linéaires allant notamment des *real time units* (secondes) vers les musical *time units* (temps, métrique,...), de nombreux travaux suggèrent d'utiliser une approche hiérarchique puisque le langage musical est lui-même structuré. En effet, l'usage d'arbres syntaxiques est idéale pour représenter le langage musical. Une méthodologie simple pour la description et l'affichage des structures musicales est présentée dans [20]. Les RT y sont évoqués comme permettant une cohésion complète de la notation musicale traditionnelle avec des notations plus complexes. Jacquemard *et al.* [21] propose aussi une représentation formelle du rythme, inspirée de modèles théoriques antérieurs issus du domaine de la réécriture de termes. Ils démontrent aussi l'application des arbres de rythmes pour les équivalences rythmiques dans [22]. 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 [19]. Les modèles de grammaire qui y sont exposés sont différents de modèles markoviens linéaires de précédents travaux.

je ne comprend pas bien l'explication. le pb est plutot vue locale (déduction de la proba d'une durée à partir de la durée précédente, par ex. dans un HMM) vs vue globale, dans une hiérarchie

RT?

techniques de réécriture appliquée à la déduction automatique, calcul symbolique

le calcul d'équiv.

citer thèse de David Rizo (Valencia)

5. https://damien.nouvels.net/fr/enseignement

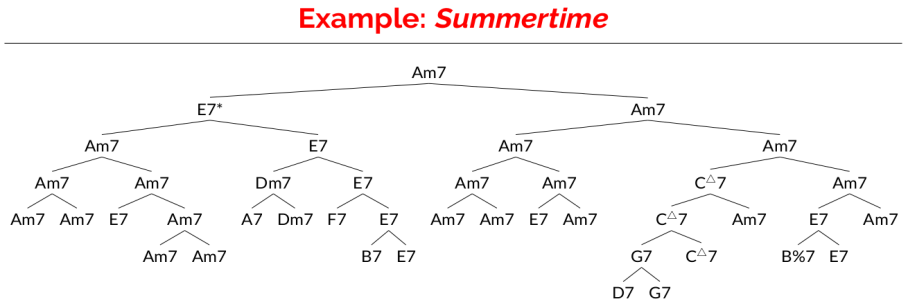


FIGURE 2.3 – arbre_jazz
Représentation arborescente d'une grille harmonique [12]

Conclusion

La plupart des travaux déjà existants sur l'ADT ont été énumérés par Wu *et al.* [14] 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 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 évènements sonores de batterie avec ceux des autres instruments dans un orchestre ou un groupe de musique [23], 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 ou MusicXML et non une partition écrite.

à ma connaissance, aucun des travaux en nADT ne produit de partition XML

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 flas ou autres ornements ;
2) les choix d'écritures spécifiques à la batterie concernant la séparation des voix et les continuations.

diff. pour production de partition (et 1 des obj. du stage) est...

latex : enumerate

CHAPITRE 3

MÉTHODES

Sommaire

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 poursuiverons avec une présentation de qparse[1], un outil de transcription qui est développé à l'Inria, l'Université de Nagoya et au sein du laboratoire Cedric au CNAM. Enfin, nous présenterons les systèmes.

plusieurs développeurs

systèmes, une représentation théorique qui permet...

3.1 La notation de la batterie

FIGURE 3.1

La figure 3.1 montre 4 figures de notes les plus courantes dont les noms et les durées sont respectivement, de gauche à droite :

— La ronde, elle vaut 4 ;

durées exprimées en unité de temps musicale, appelée le *temps*, cf. section...

4 temps

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

— La blanche, elle vaut 2 ;
— La noire, elle vaut 1 ;
— La croche, elle vaut 1/2.

plusieurs éléments

Une figure de note [1] de musique combine plusieurs critères [2] :

plutôt que wikipedia cite Dannhauser ou autre ref. F.M. ou encore Gould 2011 Behind Bars

— Une tête de note :
Sa position sur la portée indique la hauteur de la note. La tête de note peut aussi indiquer une durée.

barre verticale liée à la tête de ntoe

— Une hampe :
Indicatrice d'appartenance à une voix en fonction de sa direction et indicatrice d'une durée représentée par sa présence ou non (blanche $\neq$ ronde)

haut ou bas

— Un crochet : La durée d'une note est divisée par deux à chaque crochet ajouté à la hampe d'une figure de note.

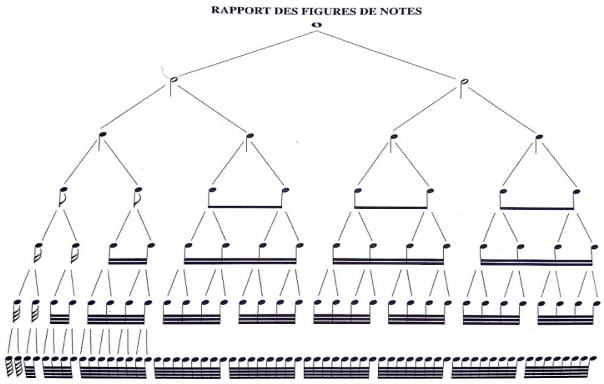


FIGURE 3.2 – Rapport des figures de notes [1]

La figure 3.2 montre les rapports de durée entre les figures de notes. Plus les durées sont longues, plus elles sont marquées par la tête de note (la note carrée fait deux fois la durée d'une ronde) 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 3.2).

ce premier paragraphe (jusqu'ici) est redondant avec §1.4 (sub. partitions). déplacer en 1.4 ? cf. proposition plus loin

Les hauteurs et les têtes de notes

Source figure 3.3 : https://www.superprof.fr/blog/composition-instrument-percussion/.

Pour la transcription, nous proposons une notation inspirée du recueil de pièces pour batterie de J.-F. Juskowiak [24] et des méthodes de batterie

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

FIGURE 3.3 – Les instruments de la batterie

Agostini [25], car nous trouvons la position des éléments cohérente et intuitive.

pour aider, tu pourrais donner une figure représentant la batterie avec le nom des instruments et abbréviation.

En effet, les hauteurs sur la portée représentent :

— 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.

— 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.

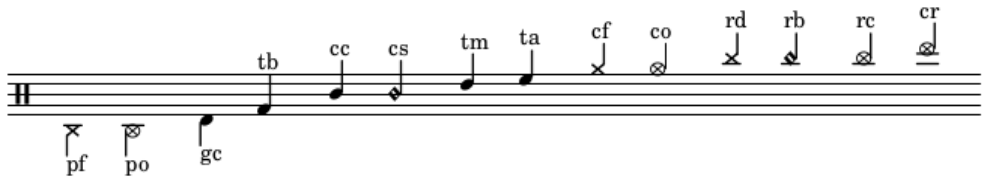


FIGURE 3.4 – Hauteur et têtes de notes

Les noms des instruments correspondant aux codes des notes de la figure 3.4 sont dans le tableau 3.1.

têtes de notes ?

Les durées

Comme nous venons de la voir, la majorité des instruments de la batterie sont représentés par les têtes des notes. Par conséquent, les symboles rythmiques concernant la tête de note ne pourront pas être utilisés. 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 batterie [26] 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.

certaines têtes de notes vides alors que leur durée n'est pas celle des blanches ?

pour clarifier cela, tu pourrais décrire en 1.4 la notation conventionnelles (piano etc) et ici uniquement ce qui est spécifique à la batterie, en expliquant les différences.

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

- Le point ;
- La liaison.

expliquer comment, par ex avec figure 3.4

Ces symboles ne seront utiles que pour l'écriture des ouvertures de charley. Le charley est le seul instrument de la batterie dont la durée est quantifiée (les cymbales attrapées à la main peuvent l'être aussi mais cela est très rare.)

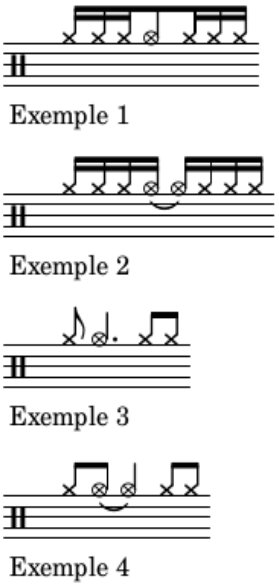


FIGURE 3.5 – Point et liaison

= la position des temps

L'écriture de la batterie doit faire ressortir la pulsation. La première chose à prendre en compte pour analyser la figure 3.5 est donc la nécessité de regrouper les notes par temps à l'aide des ligatures.

faire un "enumerate"

Exemple 1 : ouverture de charley quantifiée mais pas notes 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 plus intuitif (regroupement par temps).
En cas de nécessité de prolonger la durée d'une note au-delà de sa durée initial, et si cette note correspond à une ouverture de charley, on privilégiera la liaison.

Les silences

expliquer la notation (générale) des silebces en §1.4 ?

quantifier = noter ? ou quantifier la durée ?

Les silences sont parfois utilisés pour quantifier les ouvertures de charley. 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.

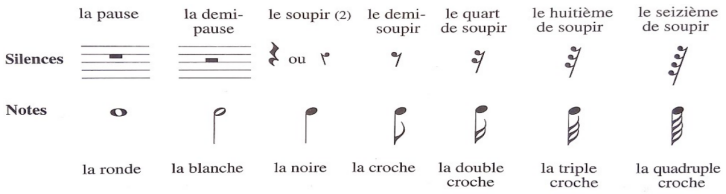


FIGURE 3.6 – Les silences

Physiquement, le charley est fermé par une pression du pied sur la pédale de charley. Dans les fichiers MIDI, cette pression est traduite par un charley joué au pied. Mais dans une vraie partition, cette écriture ne traduirait pas ce que le batteur doit penser.

pas très clair

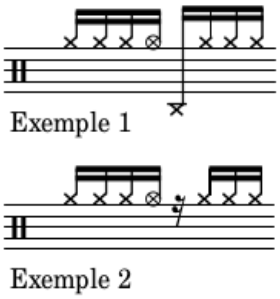


FIGURE 3.7 – Silence joué

L'exemple 1 de la figure 3.7 montre ce qui est écrit dans les données MIDI et l'exemple 2 montre ce que le batteur doit penser en lisant la partition. Il faut aussi prendre en compte l'écriture surchargée que l'exemple 1 donnerait avec une partition comprenant plusieurs voix et plusieurs instruments jouant simultanément.

Lorsqu'une note est un charley ouvert, il faudra donc prendre en compte la note suivante pour l'écriture : - Si c'est un charley fermé joué à la main ⇒ la note sera cf ;

itemize

cf?

- Si c'est un charley fermé joué au pied ⇒ la note sera un silence.

Les équivalences rythmiques

Pour les 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 pour la batterie, à part dans le cas des ouvertures de charley (voir section 3.1), les durées des notes n'ont pas d'importance. L'usage des silences pour combler la distance rythmique entre deux notes devient donc possible.

phrase alambiquée... pour prolonger la durée ?

seuls comptent les date de début de notes *onsets*.

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 à pour but de n'avoir qu'une manière d'écrire toutes les notes, que leurs têtes de notes soit modifiées ou non.

Sur la figure 3.8, 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.

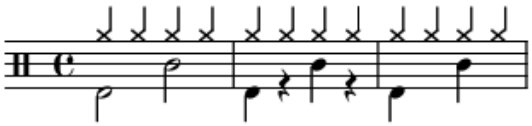


FIGURE 3.8 – Équivalence

Les voix

Les voix[3] désignent les différentes parties mélodiques constituant une composition musicale et destinées à être interprétées, simultanément ou successivement, par un ou plusieurs musiciens. En batterie, une voix est l'ensemble des instruments qui, à eux seuls, constituent une phrase rythmique et sont regroupés à l'aide des ligatures. Plusieurs écritures étant possibles pour un même rythme, on peut regrouper les instruments de la batterie par voix. Sur une portée de batterie, il existe le plus souvent 1 ou 2 voix. Sur la figure 3.9, il faudra faire un choix entre les exemples 1, 2 et 3 qui sont trois façons d'écrire le même rythme.

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.

voix : citations possibles : - "Joint Estimation of Note Values and Voices for Audio-to-Score Piano Transcription" Nakamura et al 2021 ou une des références de ce papier, par ex. [15] ou [16]. - ou thèse de Nicolas Guiomard-Kagan.

une voix est charactérisée aussi pas orientation des hampes ?

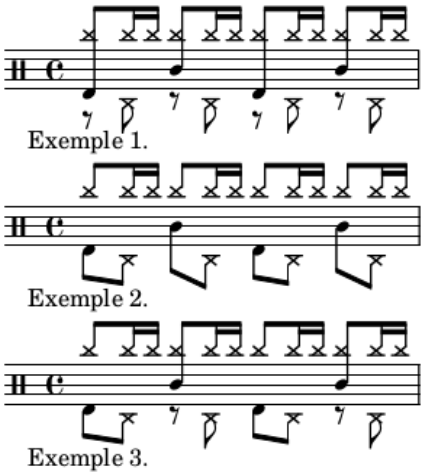


FIGURE 3.9 – Séparation des voix

Ce choix se fera en fonction des instruments joués, de la nature plus ou moins systèmatique de leurs phrasés, et des associations logiques entre les instruments dans la distribution des rythmes sur la batterie (voir la section 3.4).

3. https://fr.wikipedia.org/wiki/Voix_(polyphonie)

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]

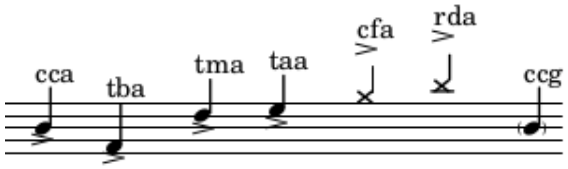


FIGURE 3.10 – Les accents et les ghost-notes

3.9 = liste des seuls "instruments" qui peuvent être accentués?

La figure 3.10 ne prend en compte que les accents que nous avons estimés nécessaires (voir la section 3.2). Les accents sont marqués par le symbole « > ». Il est positionné au-dessus des notes représentant des cymbales et en-dessous des notes représentant des toms ou la caisse-claire. Ce choix a été fait pour la partition de la figure 4.2 car elle est plus lisible ainsi, mais ces choix devront être adaptés en fonction des différents systèmes reconnus (voir la section 3.4). Par exemple, pour les systèmes jazz, les ligatures pour les toms et la caisse-claire seront dirigés vers le bas, il faudra donc mettre les symboles d'accentuation correspondants au-dessus des têtes de notes.

expliquer ce qu'est une *ghost-notes*

La dernière note de la figure 3.10 montre un exemple de ghost-notes. Le parenthésage a été choisi car il peut être utilisé sur n'importe quelle note sans changer la tête de note.

les codes de notes n'ont pas encore été présentés...

Pour les codes, on prend le code de la note et on ajoute un « a » pour un accent et un « g » pour une ghost-note. Toutes les notes de la figure 3.10 sont exposées en situation réelle dans la figure 3.11.

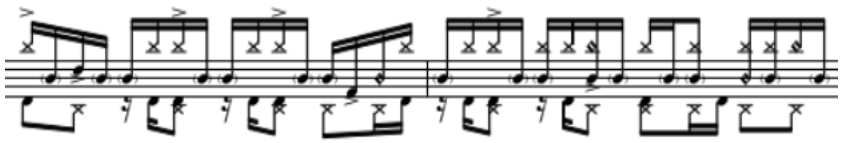


FIGURE 3.11 – Exemple pour les accentuations et les ghost-notes

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

Il existe, pour de nombreux instruments de la batterie, plusieurs samples audio associés à des pitchs. Pour cette première version, nous avons choisi de n'avoir qu'un code-instrument pour différentes variantes d'un instrument, c'est pourquoi certain code-instrument se voit attribuer plusieurs pitchs dans le tableau 3.1.

je ne comprend pas cette phrase.

il s'agit juste d'une convention de codage des instruments de la batterie en événements MIDI... que l'on prend en entrée pour la transcription

Malgré le large panel de pitchs disponible, il semblerait qu'aucun pitch ne désigne le charley ouvert joué au pied. Pourtant, dans la batterie moderne, plusieurs rythmes ne peuvent fournir le son du charley ouvert qu'avec le pied car les mains ne sont pas disponibles pour le jouer. Cela doit en partie être dû à l'utilisation des boîte 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 vélocité

La partition de la figure 4.2 a été transcrite manuellement avec lilypond par analyse des fichiers MIDI et audio correspondants.

citation lilypond

Cette transcription nous a mené aux observations suivantes :

et l'analyse d'autre fichiers MIDI ?

- Vélocité inférieure à 40 : ghost-note ;
- Vélocité supérieure à 90 : accent ;
- Pas d'intention d'accent ni de ghost-note pour une vélocité entre 40 et 89 ;

— Les accents et les ghosts-notes ne sont significatifs ni pour les instruments joués au pied, ni pour les cymbales crash.
En effet, certaines vélocités 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, nous avons 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).

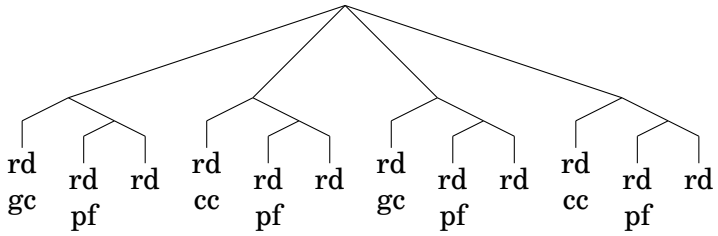
Les arbres de rythmes

Les arbres de rythmes représentent un rythme unique dont les possibilités de notation sur une partition sont théoriquement multiples.
Voici une représentation de la figure 3.9 en arbre de rythmes avec les codes de chaque instrument :

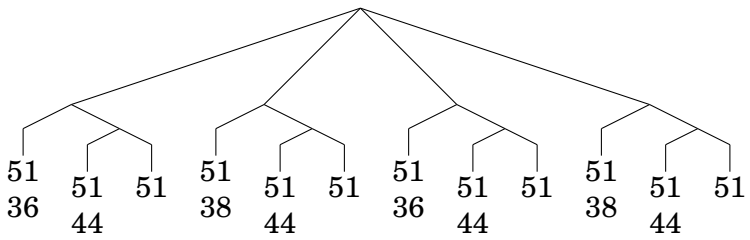
non c'est juste une représentation du rhythme, pas unique

expliquer le principe des RT : branchement = division d'intervalle temporel, feuilles = les événements musicaux commençant au début de l'intervalle). références : - Laurson "Patchwork : A Visual Programming Language", 1996. - OpenMusic : visual programming environment for music composition, analysis and research, 2011.

Fig. 3.8, ex. 1, 2 ou 3?



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



Chacun des trois exemples de la figure 3.9 est représenté par un des deux arbres syntaxiques ci-dessus.

3.3 Qparse

choisir titre plus explicite par ex. analyse syntaxique pour la transcription musicale

quantification rhythmique + structuration de partition

qparse 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 rhythmique et d'inférence de la structure de la partition à l'aide de technique de parsing / analyse syn-

La librairie Qparse [4] implémente la quantification des rythmes basée sur des algorithmes d'analyse syntaxique pour les automates arbores-

4. https://qparse.gitlabpages.inria.fr

cents pondérés. 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 XML, MEI.

grammaire ≠ automate. il faut choisir entre les 2 (pour la suite aussi)

Les principaux contributeurs sont :

- Florent Jacquemard (Inria) : développeur principal.
- Francesco Foscarin (PhD, CNAM) : construction de grammaire automatique à partir de corpus ; 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.

apprentissage

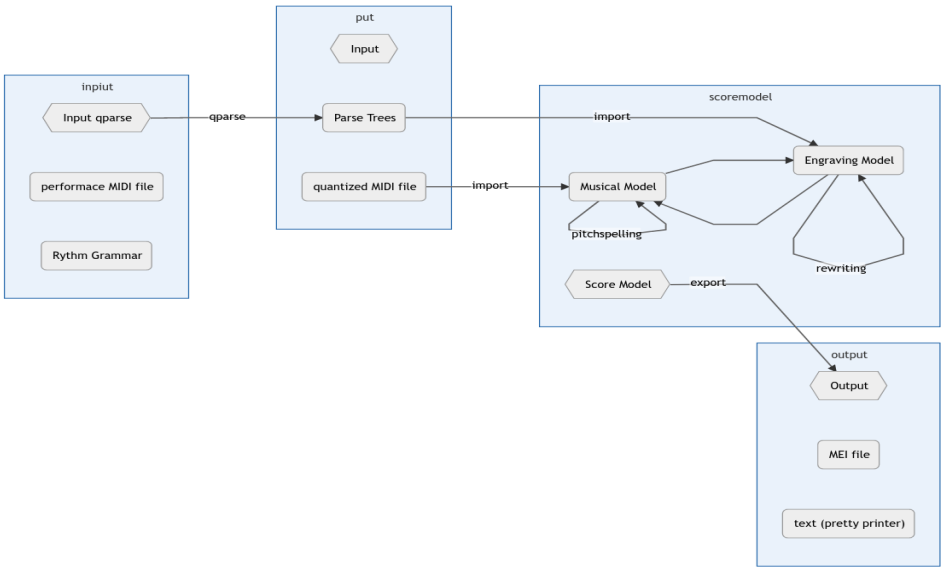


FIGURE 3.12 – Présentation de Qparse

la figure 3.11 est trop compliquée. rhythm grammar → automate d'arbres pondéré. Parse Tree → arbre syntaxique. qtz MIDI file : inutile. Score Model → représentation intermédiaire de partition. Score Model, Engr. Model : inutile. garder juste la fleche Rewriting sur S.M.

Explication des différentes étapes de la figure 3.12 [5] :

- **Input Qparse** :
  Un fichier MIDI (séquence d'événements datés (piano roll) accompagné d'un fichier contenant une grammaire pondérée) ;
- **Arbre de parsing** :
  Les données MIDI sont quantifiées, les notes de dates proches sont

5. https://gitlab.inria.fr/qparse/qparselib/-/tree/distance/src/scoremodel

alignées et les relations entre les notes sont identifiées (accords, fla, etc...) ; un arbre de parsing global est créé ;
- **Score Model** :
  - Les instruments sont identifiés dans scoremodel/import/tableImporterDrum.cpp ;
  - Réécriture 1 :
    séparation des voix ⇒ un arbre par voix ⇒ représentation intermédiaire (RI) ;
  - Réécriture 2 :
    simplification de l'écriture de chaque voix dans la RI ;
- **Output** :
  export de la partition. Plusieurs formats sont possibles (xml, mei, lilypond,...).

Plusieurs enjeux :
- Problème du MIDI avec Qparse :
  ON-OFF en entrée ⇒ 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.

3.4 Les systèmes

il faudrait expliquer là que le but est d'avoir des schemas types (= système) pour calculer la séparation en voix. = une heuristique pour éviter d'avoir à explorer une grande combinatoire. et que, une fois le système déterminé (ou sélectionné), la séparation se fait par réécriture du modèle (règles de projection + simplification)

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

Définitions

je ne comprend pas bien la définition de système : motif + gamme ou motif + gamme + texte ? la déf. des gammes n'est pas du tout claire.

Système : motif + gamme/texte
Motif : rythmes coordonnés joués avec 2 ou 3 membres en boucle (répartis sur 1 ou 2 voix)
Texte : rythme irrégulier joué avec un seul membre sur le motif (réparti sur 1 voix).

est-ce que le motif est fixe et les gammes variables ? est-ce le motif qui détermine la métrique et les voix ?

Gamme : 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.

métrique n'est pas défini. règles de réécriture non plus

Un ensemble de systèmes comprenant leur métrique et leurs règles spécifiques de réécriture sera nécessaire. Les systèmes devront être distribués

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 3.2 – Sytèmes

dans 4 grandes catégories :
Nous exposerons 3 systèmes afin d'illustrer les propos de cette section :

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

Objectif des systèmes

Les systèmes devront être matchés sur l'input MIDI afin de :

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

La partie *motif* des systèmes sera utilisée pour la **définition des métriques**. Le *motif* et la gammes des systèmes seront utilisés pour la **séparation des voix**. Les règles de **simplification** (les combinaisons de réécritures) seront extraites des voix séparées des systèmes.

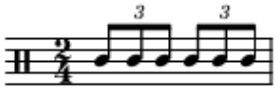
bien. il faudrait expliquer ça avant.

pas exactement. les règles de projection et simplification font la séparation en voix : à partir d'un arbre syntaxique comme celui de 3.2, elles extraient 2 arbres, chacun contenant les événements d'une seule voix

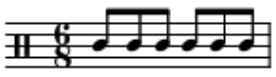
Détection d'indication de mesure

La détection de la métrique 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.

métrique ≠ signature rythmique (c'est plus général). Il aurait fallu présenter rapidement la notation des signatures rythmiques, par exemple en 1.4



Exemple 1



Exemple 2

FIGURE 3.13 – Métrique

La figure 3.13 montre deux indications de mesure 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).

Choix d'une grammaire

le lien entre grammaire et signature rythmique n'est pas clair ici. Il aurait fallu expliquer le rôle des grammaires (automates) en 3.3

Groove MIDI Data Set pas présenté

méta-données

contenu

Il faut prendre en compte l'existence potentielle de plusieurs grammaires dédiées chacune à un type de contenu MIDI. 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 métrique doit être définie avant le choix de la grammaire.
Pour les expériences effectuées avec le Groove MIDI Data Set, le style et l'indication de mesure sont récupérables par les noms des fichiers MIDI, mais il faudra par la suite les trouver automatiquement sans autres indications que les données MIDI elles-mêmes. Par conséquent, les motifs des systèmes devront être recherchés sur l'input *(fichiers MIDI)* avant le lancement du parsing, afin de déterminer la métrique en amont. Cette tâche devra probablement être effectuée en Machine Learning.

Séparation des voix

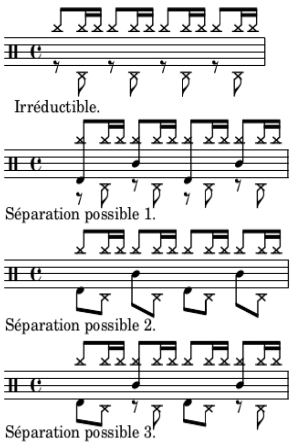


FIGURE 3.14 – Motif 4-4 binaire

les description ic sont assez techniques et difficile à suivre. avant de détailler des exemples, il faudrait décrire les objectifs et le principe de la procédure.

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 rd/pf/cc (voir 3.1) et d'un texte joué à la grosse-caisse. La première ligne de la figure 3.14 est appelée « Irréductible

» car il n'y a pas d'autre choix pertinent pour la répartition de la ride et du charley au pied. La troisième séparation proposée est privilégiée car elle répartit selon 2 voix, une voix pour les mains (rd + cc) et une voix pour les pieds (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.

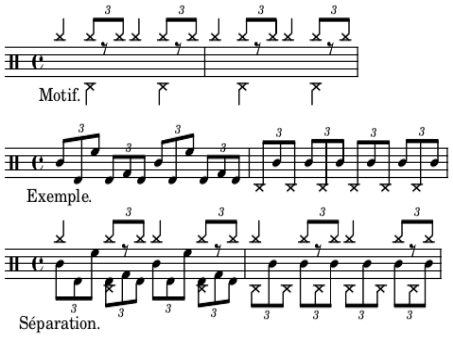


FIGURE 3.15 – Motif 4-4 jazz

Dans la plupart des méthodes, le charley n'est pas écrit car il est 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 parti pris est 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.

quel exemple ?

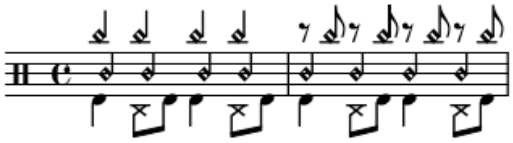


FIGURE 3.16 – Système 4-4 afro-latin

La figure 3.16 montre un exemple minimaliste de système afro-latin [26]. Ce système doit être écrit sur trois voix car la voix centrale est souvent plus complexe qu'ici (que des noirs) et la mélanger avec le haut ou le bas serait surchargé et peu lisible.

Simplification de l'écriture

Les explications qui suivent seront appuyé par une expérimentation théorique dans la section 4.4.

expérimentation théorique??

Les gammes qui accompagnent les motifs d'un système étayent toutes les combinaisons d'un système et elles permettent, combinées avec le motif d'un système, de définir les règles de simplification propres à celui-ci.
Voici les différentes étapes à suivre :

- Pour chaque gamme du système, faire un arbre de rythme représentant la gamme combinée avec le motif du système ;
- Pour chaque arbre de rythmes obtenus, séparer les voix et faire un arbre de rythme par voix ;
- Pour chaque voix (arbre 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'ouverture de charley dans les instruments joués. Nous avons discuté de ces problèmes dans le chapitre 3.
Voici quelques règles inhérentes à la simplication de l'écriture pour la batterie : Toutes les continuations (t) qui se trouvent en début de temps (figures 4.9, 4.11 et 4.12) sont transformées en silences (r) sauf si la note précédente est un charley ouvert ?

ce sont des figures et notations du chapitre suivant!

Même si on favorise l'usage des silences pour l'écart entre les notes n'appartenant pas au même temps, on les supprime systèmatiquement pour 2 notes au sein d'un même temps et favorise, une liaison si co, un point si pas co et nécessaire, un simple ajustement de la figure de note si suffisant.

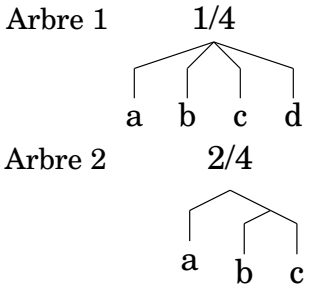


FIGURE 3.17 – Simplification

itemize

Soit l'arbre 1 de la figure 3.17 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 3.18 est attribuée :

- Si a n'est pas un co :
  ⇒ Suite 1a.

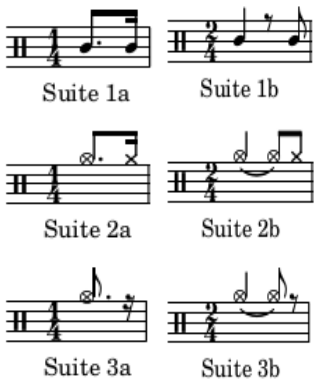


FIGURE 3.18

— Si a est un co :
  — Si d est un cf :
    ⇒ Suite 2a.
  — Si d est un pf :
    ⇒ Suite 3a : d deviens un silence (r).

Soit l'arbre 2 de la figure 3.17 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 3.18 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 deviens une liaison et c devient un silence.

Rappel :
cf = charley fermé joué à la main ;
co = charley ouvert joué à la main ;
pf = charley fermé joué au pied.

Problème : le cf et le co ne seront jamais sur la même voix que le pf... Par conséquent, les règles concernant les charleys ouverts doivent-elles être appliquées sur l'arbre de parsing de l'input ?...

Conclusion

Nous avons formalisé une notation de la batterie, modélisé cette notation pour la transcription de données MIDI en partition, nous avons décrit Qparse.
Enfin, nous avons exposé une approche de type dictionnaire (les « systèmes ») pour détecter une métrique, choisir une grammaire pondérée appropriée et énoncer des règles de séparation des voix et de simplification de l'écriture.

CHAPITRE 4

EXPÉRIMENTATIONS

Sommaire

Introduction

Dans ce chapitre, nous présenterons le jeu de données et les analyse MIDI-Audio et transcriptions manuelles.
Problèmatique : passage au polyphonique indispensable pour la suite du travail et pour l'expérimentation des systèmes rythmiques. Finir la chaîne de traitement indispensable pour obtenir des résultats chiffrés possible à évaluer.
Nous présenterons mes deux contributions principales :

- les différentes étapes de résolution du passage au polyphonique.
- l'expérimentation d'un système 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 nouveaux systèmes 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 [1] [27] (GMD) qui est un jeu de données mis à disposition par Google sous la licence Creative Commons

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

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

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 [2]
- 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 eval-session du GMD.
- Les sorties audio synthétisées ont été alignées à 2 ms près sur leur fichier MIDI.

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 ;

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.

— 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

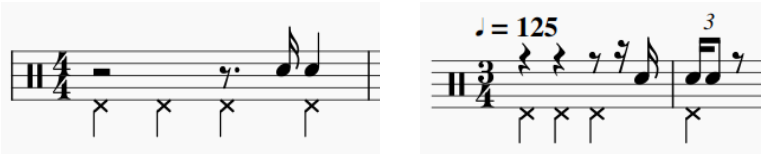
Ces analyses ont été faites dans le cadre de transcriptions manuelles à partir de fichiers MIDI et Audio du GMD.

Comparaisons de transcriptions

Pour les comparaisons de transcriptions, les transcriptions manuelles (TM) ont été éditées à l'aide de Lilypond [3] ou MuseScore [4] 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

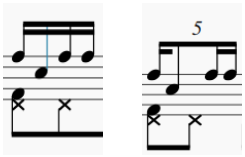


— Erreur d'indication de mesure (3/4 au lieu de 4/4) ;
— Les silences de la mesure 1 de la TA sont inutilement surchargés ;
— La noire du temps 4 de la mesure 1 de la TM est devenue les deux premières notes (une double-croche et une croche) d'un triolet sur le temps 1 de la mesure 2 de la TA.

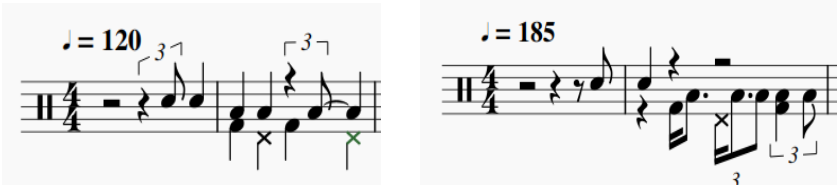
Exemple d'analyse 2

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

Transcription manuelle ⇒ Transcription automatique



Transcription manuelle ⇒ Transcription automatique

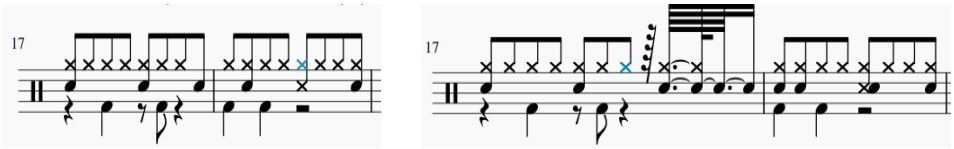


Exemple d'analyse 3

- 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 ⇒ Transcription automatique



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

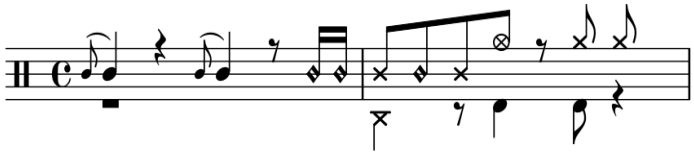
conclusion sur ces exemples

Exemple avec des flas

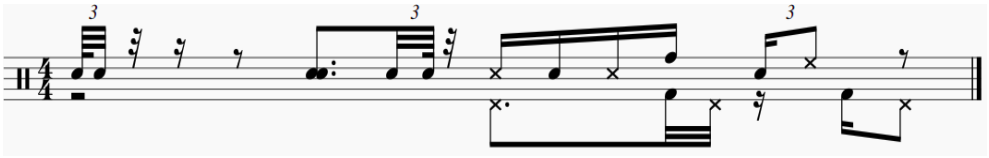
sauf erreur, les "flas" ne sont pas définis. → sections 1.4 (appogiatures) et 3.1 (flas)?

3. http://lilypond.org/
4. https://musescore.com/

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 l'air 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 3.1) 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 l'air sur le temps 2 de la mesure 2 de la TM devrait avoir le même symbole sur la TA. Idem pour les cross-sticks.

Transcription de partition

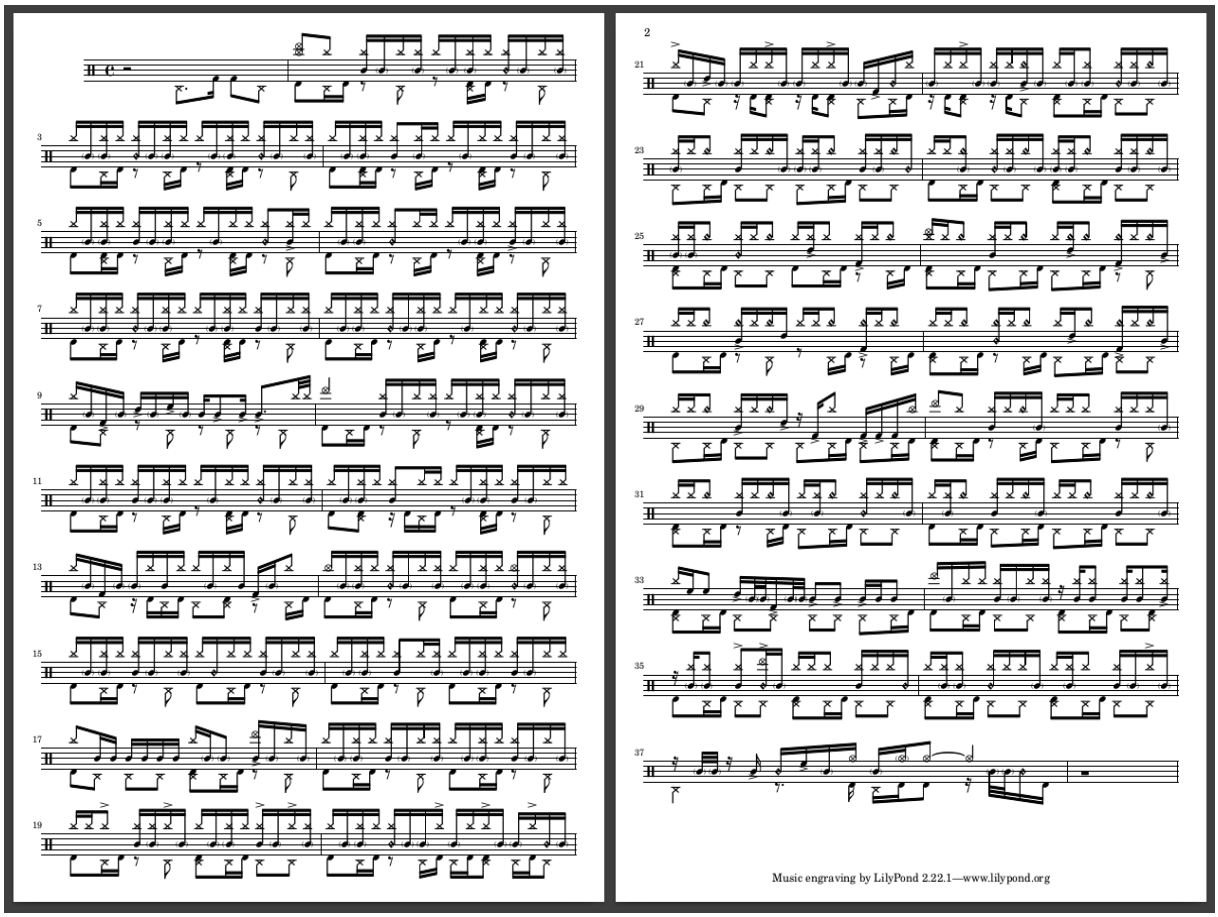


FIGURE 4.2 – Partition de référence

La figure 4.2 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). 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 des données audio et MIDI contenues dans ces fichiers a permis une analyse plus approndie des critères à relever pour chaque évènement MIDI et de la manière de les considérer dans un objectif de transcription en partition lisible pour un musicien (Voir la section 3.2).

des conclusions sur la transcription manuelle ? difficultés, durée ? nb de passes... pourquoi LilyPondet pas MuseScore ?

4.3 Transcription polyphonique par parsing (?verrou?)

<flo>Sujet de cette partie -> première problématique / contribution principale : transcription polyphonique par parsing (verrou) : jams etc</flo>

Les Jams

Les Jams permettent de passer du monophonique au polyphonique.

il faut revenir ici sur le parsing, et la notion d'alignement sur arbres syntaxiques pour définir les jams. illustrer avec les exemples précédents.

Le parsing

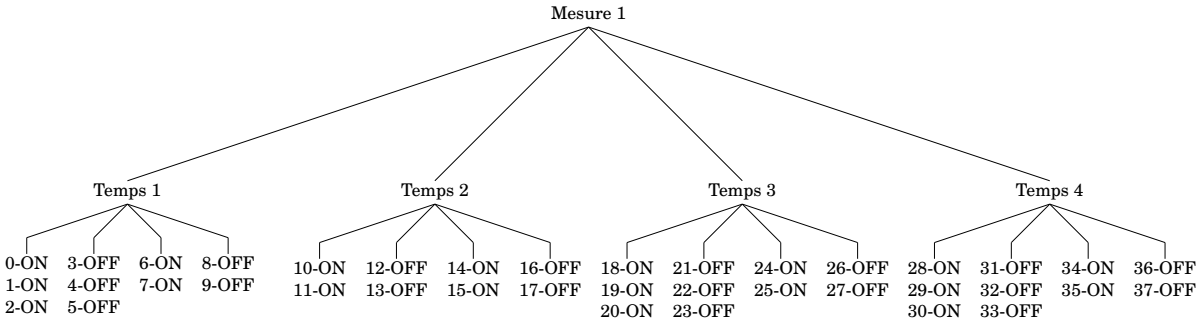
Tests effectués avec le fichier midi suivant :

Un premier test convaincant est effectué avec la grammaire suivante :

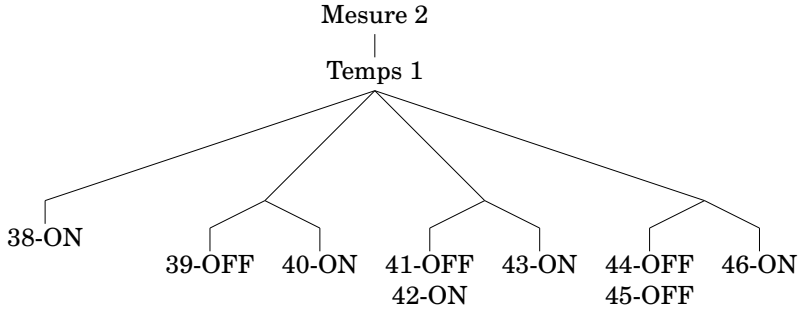
```
// bar level
0 -> C0 1
0 -> E1 1
0 -> U4(1, 1, 1, 1) 1

// 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 exemple :
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.

4.4 Expérimentation d'un système rythmique

<flo>Sujet de cette partie -> deuxième problèmatique / contribution principale : réécriture, pour séparation en voix et simplification, aidée (guidée) par système rythmique.</flo> Cette expérimentation théorique, basée sur la partition de référence de la figure 4.2, montre le procédé de création d'un *système rythmique* et des règles qui en découlent (signature rythmique, choix de grammaire, règles de séparation des voix et de simplification de l'écriture). Le *système rythmique* devra ensuite être implémenté pour appliquer des tests qui seront effectués, dans un premier temps, sur la partition de référence.

Le titre est contradictoire, et l'explication pas très claire

Motifs et gammes

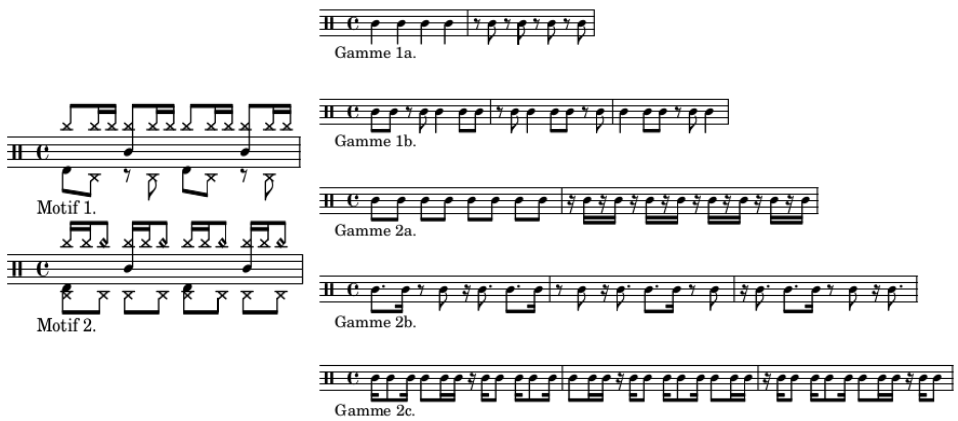


FIGURE 4.3 – Motifs et gammes

Motifs

À partir de la partition de référence, les deux motifs de la figure 4.3 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 mesures 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 4.3 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 l'air. 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 l'air. Ces deux possibilités sont combinées de manière circulaire dans les lignes 4 et 5 qui contiennent chacunes 3 mesures.

système rythmiques — motifs et gammes combinés

Pour la suite de l'expérimentation théorique, nous utiliserons le motif 1 de la figure 4.3.

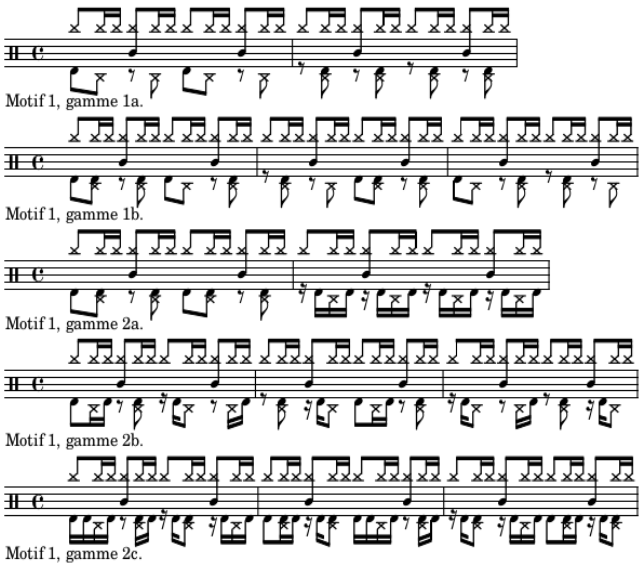


FIGURE 4.4 – Partition d'un système rythmique en 4/4 binaire

Représentation du système rythmique en arbres de rythmes

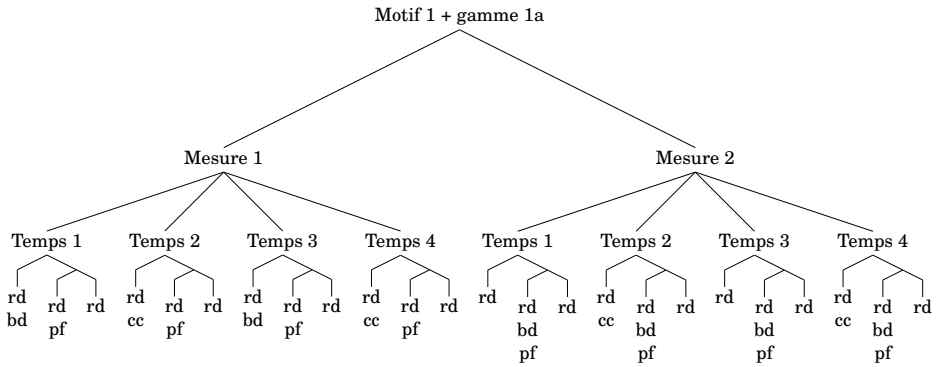


FIGURE 4.5 – Arbre de rythme — système rythmique

L'arbre de la figure 4.5 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 4.4. 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 régles du système rythmique.

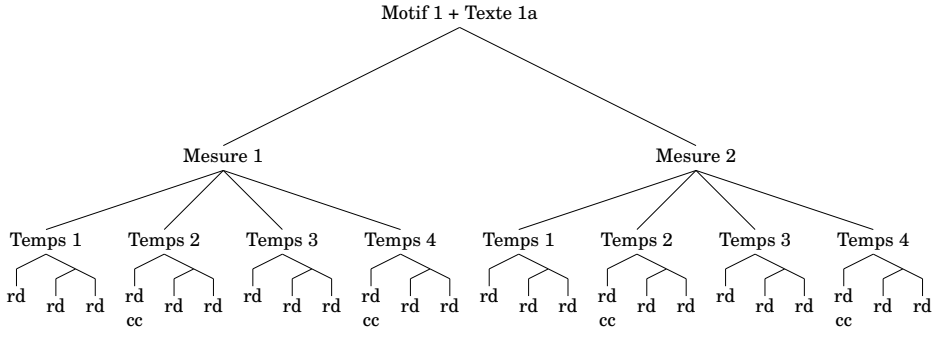


FIGURE 4.6 – Arbre de rythme — voix haute

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

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

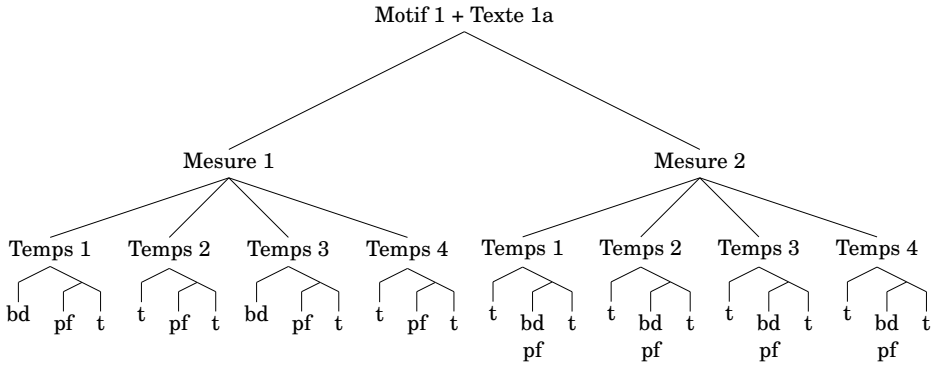


FIGURE 4.7 – Arbre de rythme — voix basse

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 (rendre la pulse visuelle).

Pour les figures ci-dessous :

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

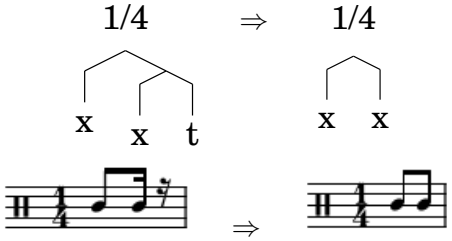


FIGURE 4.8

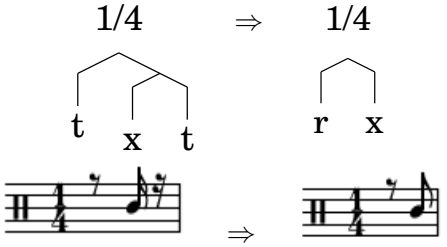


FIGURE 4.9

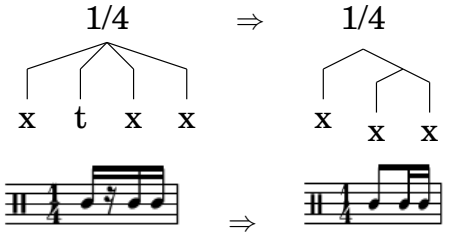


FIGURE 4.10

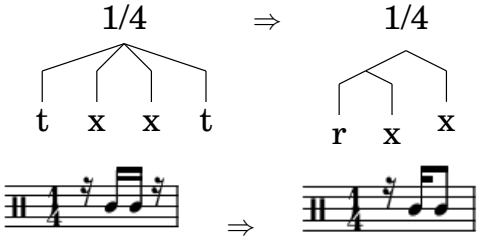


FIGURE 4.11

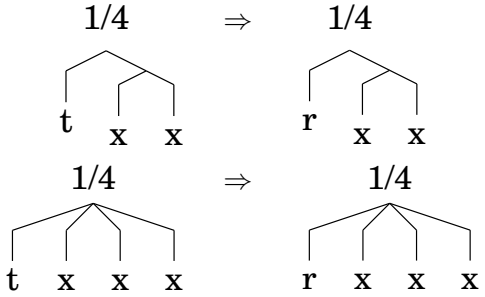


FIGURE 4.12

Ces règles ont été tirées de l'ensemble des arbres du système 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 ce système rythmique mais lorsqu'il y a des ouvertures de charley, cela n'est pas toujours applicable. Ce problème est évoqué de le chapitre 3.

⇒ **Objectif de cette expérimentation théorique** :

La méthode des *système rythmiques* étant basée sur une approche dictionnaire, cette expérimentation théorique a pour but d'orienter la recherche d'autres système rythmiques par observation du jeu de données et de montrer comment les construire pour agrandir la base de connaissance de Qparse pour l'ADT.

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 améliorer 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 été aussi été obtenu (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 avancés 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 au arbres de parsing, nous avons traité le polyphonique (identification des regroupements de notes [5]) ⇒ Quelques arbres ont été obtenus sur des exemples simples ([6])
- 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 système rythmique basique réprésentatif 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.

5. fla ou accords entre autres...
6. exemple de 2 mesures, voir ...

L'évaluation est-elle automatique ou manuelle ?
Possibilité d'un export lilypond en arbre pour comparer l'ouput 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 + ouput 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.

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 évènements 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 [7] 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, 30 et 36.

[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, 22 et 63.

[12] Daniel Harasim, Christoph Finkensiep, Petter Ericson, Timothy J O'Donnell, and Martin Rohrmeier. The jazz harmony treebank. – Cité pages 14 et 27.

[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, 23 et 27.

[15] Moshekwa Malatji. Automatic music transcription for two instruments based variable q-transform and deep learning methods, 10 2020. – Cité page 22.

[16] 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 24.

[17] 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 24.

[18] 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 24 et 26.

[19] 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 24 et 26.

[20] 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 26.

[21] Florent Jacquemard, Pierre Donat-Bouillud, and Jean Bresson. A Term Rewriting Based Structural Theory of Rhythm Notation. Re-

search report, ANR-13-JS02-0004-01 - EFFICACe, March 2015. – Cité page 26.

[22] 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 26.

[23] 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 27.

[24] J.-F. Juskowiak. *Rythmiques binaires 2*. Alphonse Leduc, Editions Musicales, 175, rue Saint-Honoré, 75040 Paris, 1989. – Cité page 30.

[25] Dante Agostini. *Méthode de batterie, Vol. 3*. Dante Agostini, 21, rue Jean Anouilh, 77330 Ozoir-la-Ferrière, 1977. – Cité page 31.

[26] 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 32 et 43.

[27] 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 47.