

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

<b>Liste des figures</b>	<b>5</b>
<b>Liste des tableaux</b>	<b>5</b>
<b>Introduction</b>	<b>7</b>
<b>I Contexte général</b>	<b>11</b>
<b>1 État de l'art</b>	<b>13</b>
1.1 Introduction . . . . .	13
1.2 Contenu . . . . .	13
1.3 Conclusion . . . . .	14
<b>2 Transcription automatique</b>	<b>15</b>
2.1 Introduction . . . . .	15
2.2 Transcription MIDI vers partition . . . . .	15
2.3 Conclusion . . . . .	16
<b>3 Méthodes</b>	<b>17</b>
3.1 Introduction . . . . .	17
3.2 Représentations, systèmes et réécriture . . . . .	19
3.3 Conclusion . . . . .	26
<b>II Expérimentations</b>	<b>27</b>
<b>4 Corpus</b>	<b>29</b>
4.1 Introduction . . . . .	29
4.2 Contenu . . . . .	32
4.3 Conclusion . . . . .	43
<b>5 Résultats</b>	<b>45</b>
5.1 Introduction . . . . .	45
5.2 Contenu . . . . .	45
5.3 Conclusion . . . . .	45
<b>6 Discussion</b>	<b>47</b>
6.1 Introduction . . . . .	47
6.2 Contenu . . . . .	47
6.3 Conclusion . . . . .	47
<b>Conclusion générale</b>	<b>49</b>

**Bibliographie****51**

## **LISTE DES FIGURES**

## **LISTE DES TABLEAUX**

3.1	Pitchs et instruments . . . . .	20
3.2	Vélocité et nuances . . . . .	20

*En règle générale, l'introduction et la conclusion sont les deux sections de contenu à ne pas être numérotées. Idéalement, chaque chapitre commence par une introduction rapide et se termine par une conclusion rapide pour aider le lecteur à mémoriser et comprendre ce qui a été fait.*

# INTRODUCTION

*Introduction : présentation générale du contexte et de la problématique traitée, plan suivi dans le mémoire.*

## Présentation générale

### Le contexte

Ce mémoire de recherche, effectué en parallèle d'un stage à l'Inria dans le cadre du master de traitement automatique des langues de l'Inalco, contient une proposition d'amélioration d'une chaîne de traitement d'ADT de bout en bout.

Même si ce sujet ne traite pas directement de langues naturelles, son objet est l'écriture automatique de partitions de musique à partir de données audios. Il nécessite donc la manipulation d'un langage musical codifié avec une grammaire (solfège, durées, nuances, volumes) et soulève de nombreuses problématiques proches de la reconnaissance de la parole.

*Notes discussion Damien :*

**Musique**  $\Rightarrow$  langue ou langage ? ;

**Partition musicale**  $\Rightarrow$  Manière d'écrire la musique. . .

La chaîne de traitement peut-être séparée en deux grandes parties :

- Le traitement du signal à partir d'enregistrements audios de performances de batteurs et l'écriture des données sur des fichiers MIDI.
- La transformation des fichiers MIDI en partition de batterie (chaque fichier MIDI étant une partition)

Le sujet du stage se concentre sur la deuxième partie de cette chaîne et le sujet du mémoire est une proposition d'amélioration sur cette partie de la chaîne.

### Sujet du stage

La présente proposition s'intéresse à ce dernier problème, et plus précisément à :

1. l'étude de modèles de langage (LM) incorporant certaines informations musicales de haut niveau nécessaires à la génération de partitions de qualité. On devrait en particulier considérer des hiérarchies d'événements de batterie induisant des placements temporels cohérents et se prêtant à des notations rythmiques faciles à lire pour un batteur entraîné; voir [1] pour des modèles structurés en arbre basés sur la théorie formelle du langage, que nous développons dans le contexte d'outils AMT plus généraux.

2. l'intégration de ces LM avec l'état de l'art des modèles acoustiques (AM) et

des méthodes pour les tâches de traitement du signal ci-dessus. Cela nécessite la prise en compte du contexte musical et des informations musicales de haut niveau des LM en plus des caractéristiques acoustiques de bas niveau ci-dessus.

En outre, certaines expériences seront menées sur la base d'ensembles de données publiques, afin d'évaluer l'approche intégrée. Elles devraient couvrir certains cas de motifs rythmiques complexes se chevauchant.

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

## Sujet du mémoire

Le mémoire propose de rechercher de rythmes génériques (*motifs*) en amont dans la chaîne de traitement. Les *motifs* sont prédéfinis avec des combinaisons possibles (*gammes*) qui leur sont associées. Ces *motifs* et leur *gammes* respectives sont appelés *systèmes*.

L'usage des *systèmes* a pour objectif de fixer des choix le plus tôt possible dans la chaîne de traitement afin de simplifier le reste des calculs en éliminant une grande partie d'entre eux. Ces choix concernent notamment la métrique, la séparation des voix ainsi que les règles de réécriture.

## Problématique traitée

En entrée : midi (séquence d'événements datés (piano roll) accompagné d'une grammaire pondérée)

⇒ parsing

⇒ global parsing tree

⇒ RI (Représentation Intermédiaire) arbres locaux par instruments

⇒ Sortie (xml, mei, lilypond, . . . )

Minimiser la distance entre le midi et la représentation en arbre.

Le but du stage est d'améliorer qparse, un outil de transcription et d'écriture automatique de la batterie (entre autre)

Le sujet de ce mémoire est de proposer une tâche de reconnaissance du regroupement des notes par les ligatures dans l'écriture de la batterie.

Pour cela, nous utiliserons la logique des systèmes (selon la définition agostinienne).

⇒ Motif répétitif de plusieurs instruments coordonnées accompagnés d'un texte varié joué par un autre instrument de la batterie.

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

Nous travaillerons aussi sur la détection de répétitions sur plusieurs mesures



afin de pouvoir corriger des erreurs sur une des mesures qui aurait dû être identique au autres mais qui présente des différences.

**Plan suivi dans le mémoire**

- écouter le dataset groove
- observer la structure midi
- décrire la notation de la batterie et transcrire manuellement pour comparer l'input et l'output idéal d'un point de vu théorique.
- Détecter les systèmes dans le dataset.
- Faire des règles de réécriture (rewriting/simplification) et de séparation des voix (reconnaissance de systèmes)



**Première partie**

**Contexte général**



# ÉTAT DE L'ART

## Sommaire

1.1	Introduction . . . . .	13
1.2	Contenu . . . . .	13
1.3	Conclusion . . . . .	14

*L'état de l'art (chapitre 1) : les articles qui traitent du même sujet que vous, présentés en un tout cohérent (extraire de chaque article lu les points essentiels et présenter dans ce chapitre le résultat de ces lectures en regroupant les articles par point essentiel)*

## 1.1 Introduction

Dans ce chapitre, nous présentons les différentes avancées qui ont déjà eues lieux dans le domaine de la transcription.

## 1.2 Contenu

L'objectif de la transcription automatique de la musique (AMT) [3] est de convertir la performance d'un musicien en notation musicale - un peu comme la conversion de la parole en texte dans le traitement du langage naturel. Elle est considérée comme l'un des problèmes de recherche les plus anciens et les plus difficiles dans le domaine de la recherche d'information musicale (MIR).

Le cas de la transcription de la batterie (DT) est très particulier puisqu'il s'agit d'instruments sans hauteur, d'événements avec (presque) aucune durée et de notations spécifiques. Il a été la source de nombreuses études MIR, voir [2] pour un aperçu. La plupart de ces travaux se concentrent sur des méthodes de calcul pour la détection d'événements sonores de batterie à partir de signaux acoustiques, et sur l'extraction de caractéristiques de bas niveau telles que la classe d'instrument et le moment de l'apparition du son (peak picking). Cependant, très peu d'entre eux ont abordé la tâche de générer une notation musicale (rythmique) lisible à partir des caractéristiques ci-dessus, une étape cruciale dans un contexte musical et loin d'être triviale.

**Automatic music transcription : Challenges and future directions [3]**  
(introduction[3])

**Les applications de l'AMT ont aussi de la valeur dans les domaines oraux qui manquent de partition (jazz, pop, (et donc batterie, note perso)**  
(abstract [3])

Les différents travaux existant se préoccupent plus de la transcription à partir de l'audio en passant par le traitement du signal.

Les humains sont encore meilleur que les machines et la précisions à l'air d'avoir atteint sa limite.

Analyse des limites des méthodes courantes et identification des directions prometteuses.

Les modèles généraux utilisés ne traitent pas correctement la riche diversité des signaux musicaux.

2 moyens pour surmonter cela :

- adapter algo pour des cas d'utilisations spécifiques.
- utiliser les approches semi-automatiques.

**La richesse des partitions musicales et des données audio correspondantes, désormais disponibles, constitue une source potentielle de données d'apprentissage, grâce à l'alignement forcé des données audio sur les partitions, mais l'utilisation à grande échelle de ces données n'a pas encore été tentée.**

D'autres approches prometteuses incluent l'intégration d'informations provenant de plusieurs algorithmes et de différents aspects musicaux.

**A Review of Automatic Drum Transcription**[\[2\]](#)

### 1.3 Conclusion

Conclusion de ce chapitre.

# TRANSCRIPTION AUTOMATIQUE

## Sommaire

2.1	Introduction . . . . .	15
2.2	Transcription MIDI vers partition . . . . .	15
2.2.1	Les données MIDI . . . . .	15
2.3	Conclusion . . . . .	16

## 2.1 Introduction

Dans ce chapitre, nous expliquerons le processus général de la transcription automatique.

Nous ne parlerons que de la deuxième partie de la chaîne de traitement allant des données midi vers l'audio.

## 2.2 Transcription MIDI vers partition

### 2.2.1 Les données MIDI

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

<https://en.wikipedia.org/wiki/MIDI>

## **2.3 Conclusion**

Conclusion de ce chapitre.



## MÉTHODES

### Sommaire

3.1	Introduction . . . . .	<b>17</b>
3.1.1	Chaîne de traitement . . . . .	17
3.2	Représentations, systèmes et réécriture . . . . .	<b>19</b>
3.2.1	La notation de la batterie . . . . .	19
3.2.2	La notation midi de la batterie . . . . .	19
3.2.3	Les systèmes . . . . .	22
3.3	Conclusion . . . . .	<b>26</b>

*Méthodes (chapitre 3) : les méthodes appliquées, avec le détail des expériences réalisées (différentes configurations);*

*Corpus (chapitre 4) : le corpus utilisé (caractéristiques, pré-traitements appliqués)*

## 3.1 Introduction

Dans ce chapitre...

### 3.1.1 Chaîne de traitement

- Reconnaître un motif (système) dans une partition (un fichier midi)  
Sur une mesure de l'input  $\Rightarrow$  Motif (système) reconnu : true ou false
- Si true :
  - Séparer les voix
  - Simplifier l'écriture de chaque voix

Chaîne de traitement :

```
if match(system_parse_tree(system + pitch), input_midi_parse_tree)
  then voice_split;
  for voice in voice_split :
    simplification voice
```

### Les systèmes en batterie

SYSTÈME  $\Rightarrow$  MOTIF (2-3 instruments sur 1-2 voix, joué en boucle) + TEXTE (1 instrument sur 1 voix, irrégulier)

ex : système afro-cubain, trois voix. Proposition pour la détection de la direction des hampes et pour les ligatures (regroupement des notes et séparation des voix.)

— **Les systèmes :**

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

Définition d'un système :

En cas de système, les ligatures forment deux voies :

- Le texte ;
- Le système.

*Mettre des exemples de différents systèmes.*

— **Les moulins :**

Lorsqu'il y a plus d'une voie, ils sont prioritaires pour les ligatures.

*Mettre des exemples.*

## Gestion des silences et des têtes de notes

Rythme tree (RT)

- Le symbole "-" est une continuation (liaison) mais pour une partition de batterie, ça serait un silence par défaut sauf peut-être pour les ouvertures de charley et éventuellement les cymbales ou un tom basse qui résonne.

⇒ Question de la (notation noir + silence) vs blanche.

⇒ On privilégierait (noir + silence) puisque les symboles « x » des cymbales ne peuvent pas porter d'indication de durée dans la tête de notes.

Les 3 parties d'une note :

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

source : [https://fr.wikipedia.org/wiki/Note\\_de\\_musique](https://fr.wikipedia.org/wiki/Note_de_musique)

## Définition des têtes de notes et des hauteurs

### Proposition de définition d'un standard de départ

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

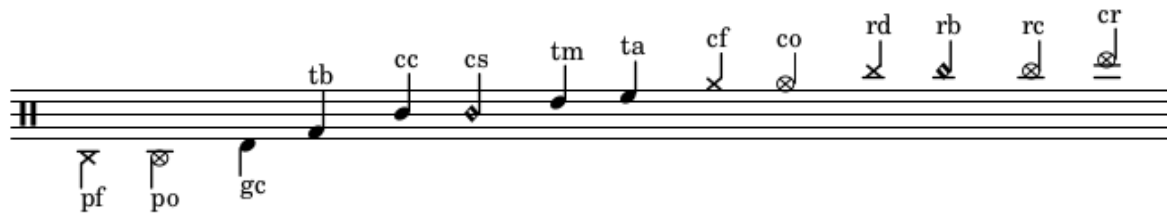
Cette symétrie s'opère en trois dimensions :

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

## 3.2 Représentations, systèmes et réécriture

### 3.2.1 La notation de la batterie

#### La hauteur des notes et les têtes de notes



partition 1

#### Les nuances



partition 2

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

#### La séparation des voix



partition 3

### 3.2.2 La notation midi de la batterie

#### Représentation symbolique (MIDI)

.....

## Les pitches et la vélocité

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

TABLE 3.1 – Pitches et instruments

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

TABLE 3.2 – Vélocité et nuances

Pas de charley pied ouvert. . . Nous ne prendrons en compte la vélocité que pour la cc, les toms et les cymbales jouées aux mains. Les nuances de grosse caisse et charley aux pieds sont le plus souvent insignifiantes, elles ne sont marquées sur le figure qu'à titre indicatif.

Si la vélocité est en dessous de 40, il s'agit de ghost-notes : la tête de note devra être entouré de parenthèses et le suffixe *p* (*piano*) devra être ajouté au codes de l'instrument. (Voir ccp ci-dessus.)

Si la vélocité est au dessus de 90, il s'agit de notes accentuées : le symbole « > » et le suffixe *f* (*forte*) devra être ajouté au codes de l'instrument. (Voir ccf ci-dessus.)

Lorsque la vélocité va de 40 à 89, on considèrera le volume comme normal et aucun symbole supplémentaire ne sera ajouté à la note.

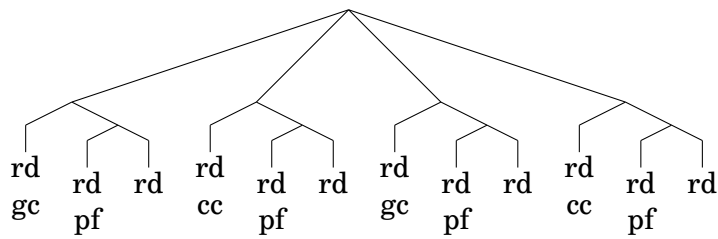
L'instrument qui sera difficile à placer sera la caisse claire car elle ne sera pas toujours affiliée aux mêmes instruments.

## Les dilemmes

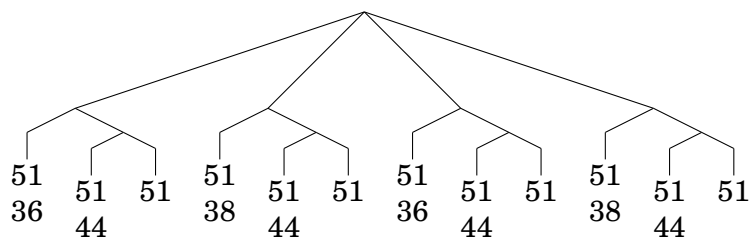
Le charley de pitch 46 est considéré comme le charley ouvert joué à la main sur le haut de la cymbale mais souvent, ça correspond au geste « tranche-olive » de la baguette lorsque le batteur accentue avec la tranche et joue moins fort avec l'olive sur le plat de la cymbale. Je vais dans un premier temps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

### Représentations en arbres

Voici une représentation de la *partition 3* en arbre de rythme avec les codes de chaque instrument :



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



Cet arbre représente un rythme unique dont les possibilités de notation sur une partition sont théoriquement multiples. (Voir *partition 3*).

### 3.2.3 Les systèmes

#### Définition

Système = motif + texte

motif = rythmes coordonnés joués avec 2 ou 3 membres en boucle

texte = rythme irrégulier joué avec un seul membre sur le motif.

Quatre systèmes standards :

- binaire
- ternaire (shuffle, afro, rock)
- jazz
- afro-cubain

#### Utilité

- Séparation des voix
- Définir une métrique
- Conditionner des règles spécifiques de réécriture

Créer un ensemble de systèmes :

- 4/4 binaire FAIT
- jazz vs ternaire(12/8) EN COURS...
- afro-cubain
- Tout transcrire avec lilypond et en arbres d'analyse syntaxique.
- Créer les arbres de voix séparées.
- Écrire les règles de réécriture.
- Créer les arbres de voix séparées simplifiés (rewriting).

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

## Pour la séparation des voix

### Motif 4-4 binaire

The notation shows a motif on a grand staff (treble and bass clefs) in common time (C). The motif consists of a sequence of eighth notes on the treble staff and eighth notes on the bass staff. The first staff is labeled 'Irréductible.' and shows the motif as a single unit. The second staff is labeled 'Séparation possible 1.' and shows the motif split into two groups of four eighth notes each. The third staff is labeled 'Séparation possible 2.' and shows the motif split into two groups of four eighth notes each, with the first group on the treble staff and the second group on the bass staff. The fourth staff is labeled 'Séparation possible 3.' and shows the motif split into two groups of four eighth notes each, with the first group on the treble staff and the second group on the bass staff, but with a different grouping of the notes.

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

### Motif 4-4 jazz

The notation shows a motif on a grand staff (treble and bass clefs) in common time (C). The motif consists of a sequence of eighth notes on the treble staff and eighth notes on the bass staff. The first staff is labeled 'Motif.' and shows the motif as a single unit. The second staff is labeled 'Exemple.' and shows the motif split into two groups of four eighth notes each. The third staff is labeled 'Séparation.' and shows the motif split into two groups of four eighth notes each, with the first group on the treble staff and the second group on the bass staff.

Dans la plupart des méthodes, le charley n'est pas écrit car considéré comme évident en jazz traditionnel. Ce qui facilite grandement l'écriture : la ride et les crash sur la voix du haut et le reste sur la voix du bas. Ici, le partie prit et de tout écrire. Dans l'exemple ci-dessus, les mesures 1 et 2 combinées avec le *motif* de la première ligne, sont des cas typiques de la batterie jazz. Tout mettre sur la voix haute serait surchargé. De plus, la grosse caisse entre très souvent dans le flot des combinaisons de toms et de caisse claire et son écriture séparée serait inutilement compliquée et peu intuitive pour le lecteur. Le choix de séparation sera donc de laisser les cymbales en haut et toms, caisse-claire, grosse-caisse et pédale de charley en bas.

**Système 4-4 afro-cubain**

Séparation sur trois voix.

**Pour la reconnaissance de la métrique**

*12/8 vs 4/4 ternaire*

**Motif 12/8****Pour les règles de réécriture**

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



## Construction des systèmes pour les expérimentations

### La réécriture des événements MIDI pour la batterie

Basé sur [4] et sur [5]

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

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

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

#### Exemples :



A1 — Évènement MIDI.



A2 — Réécriture.



B1 — Évènement MIDI.



B2 — Réécriture.

#### Exemples à écrire en arbre :

- SI (pas pf) ET (note sur un temps suivie de note en l'air) :  
⇒ (Temps1 : Note pertinente) + (Temps2 : Silence pertinent + Note pertinente.)
- Si (po ou co) déborde sur le temps suivant :  
⇒ Liaison car marchera dans tous les cas même la où le point ne marchera pas (voir A2).
- Une blanche sera écrite noir + soupir.

### Les règles de réécriture

$$\begin{array}{c} \frac{2}{8} \\ \diagup \quad \diagdown \\ x \quad tie \end{array}$$

$$\begin{array}{c} \frac{2}{8} \\ | \\ x \end{array}$$

$$\begin{array}{c} \frac{1}{4} \\ \diagup \quad \diagdown \\ x \quad tie \end{array}$$

$$\begin{array}{c} \frac{1}{4} \\ \diagup \quad \diagdown \\ x \quad r \end{array}$$

### 3.3 Conclusion

Conclusion de ce chapitre.

**Deuxième partie**

**Expérimentations**



## CORPUS

## Sommaire

---

4.1	Introduction . . . . .	29
4.2	Contenu . . . . .	32
4.2.1	Expérience 1 - 4/4 binaire . . . . .	32
4.2.2	Expérience 2 . . . . .	40
4.2.3	squant : parsing du fichier midi . . . . .	41
4.3	Conclusion . . . . .	43

---

## 4.1 Introduction

Dans ce chapitre... **groove MIDI dataset**

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



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

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

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

## Les métadatas :

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

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

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

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

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

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

## 4.2 Contenu

### 4.2.1 Expérience 1 - 4/4 binaire

#### Partition de référence pour l'ouput

Music engraving by LilyPond 2.22.1—www.lilypond.org

### Systèmes recherchés

Textes :

Débit croches.



Débit doubles-croches.





Motifs :

Systemes resultants :

Motif 1, texte 1a.

Motif 1, texte 1b.

Motif 1, texte 2a.

Motif 1, texte 2b.

Motif 1, texte 2c.

Motif 2, texte 1a.

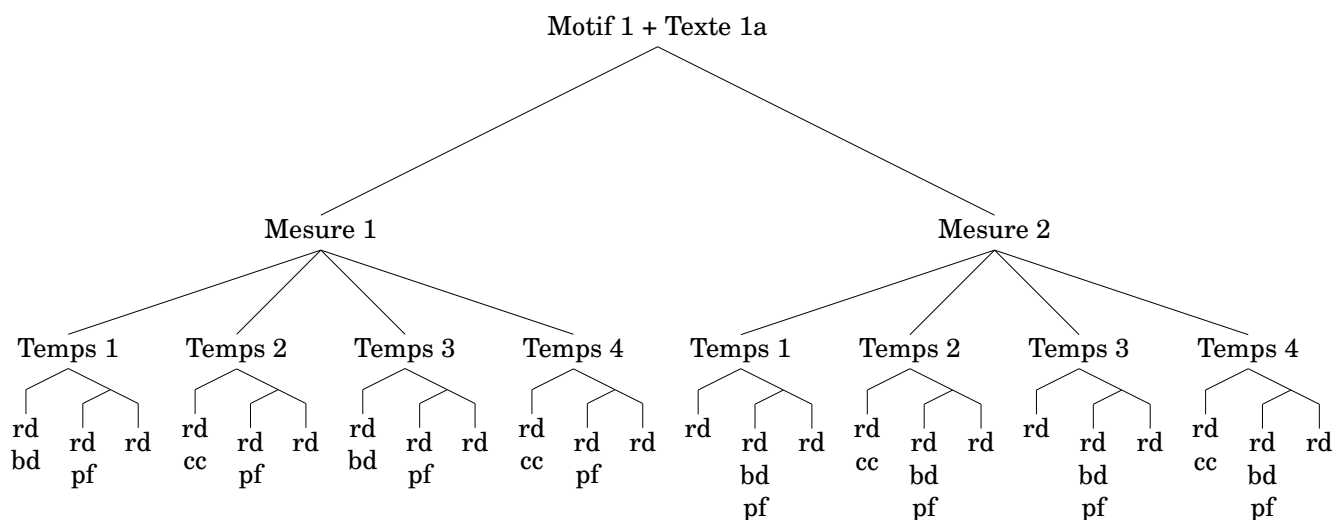
Motif 2, texte 1b.

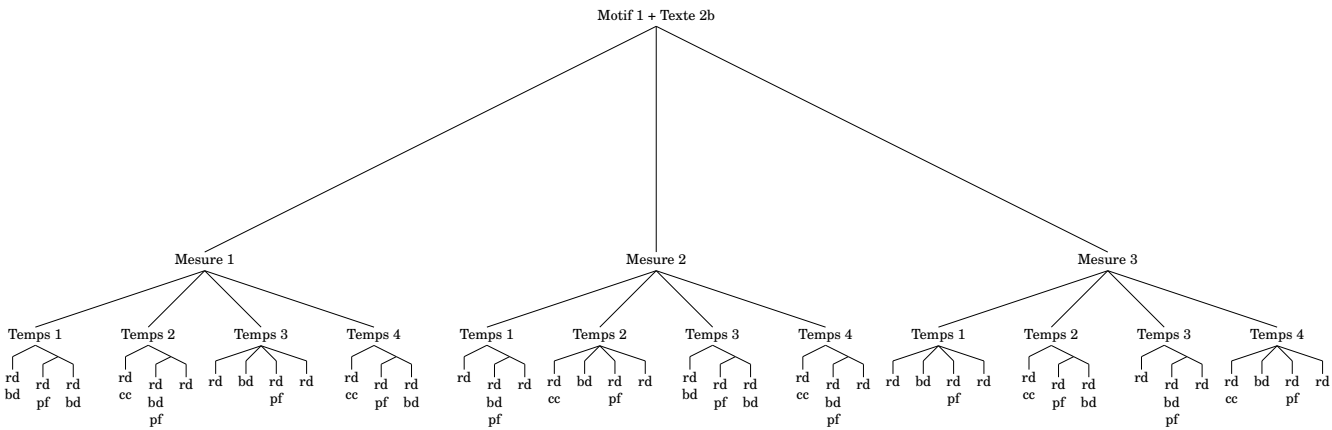
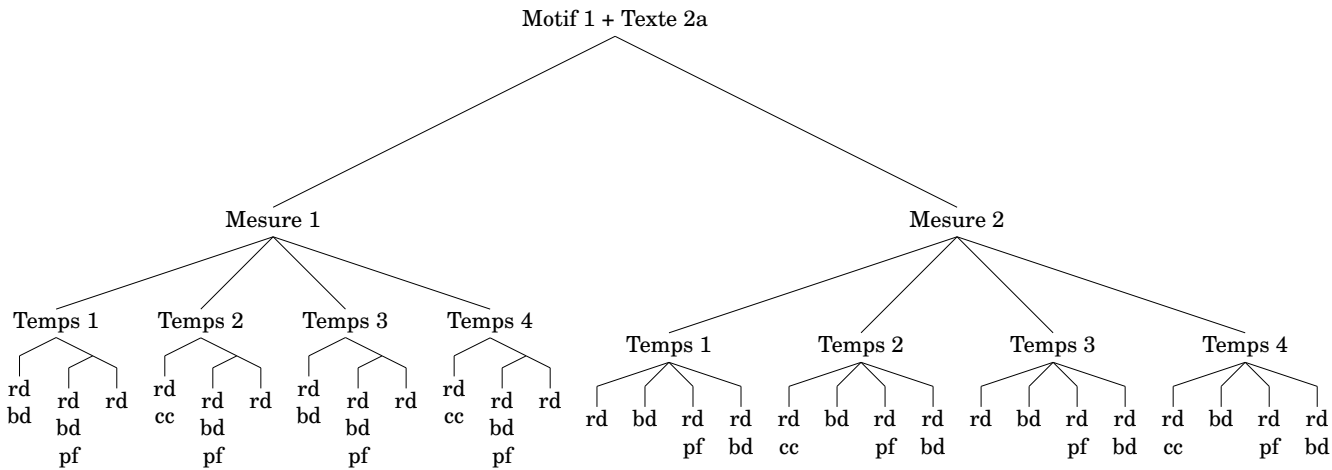
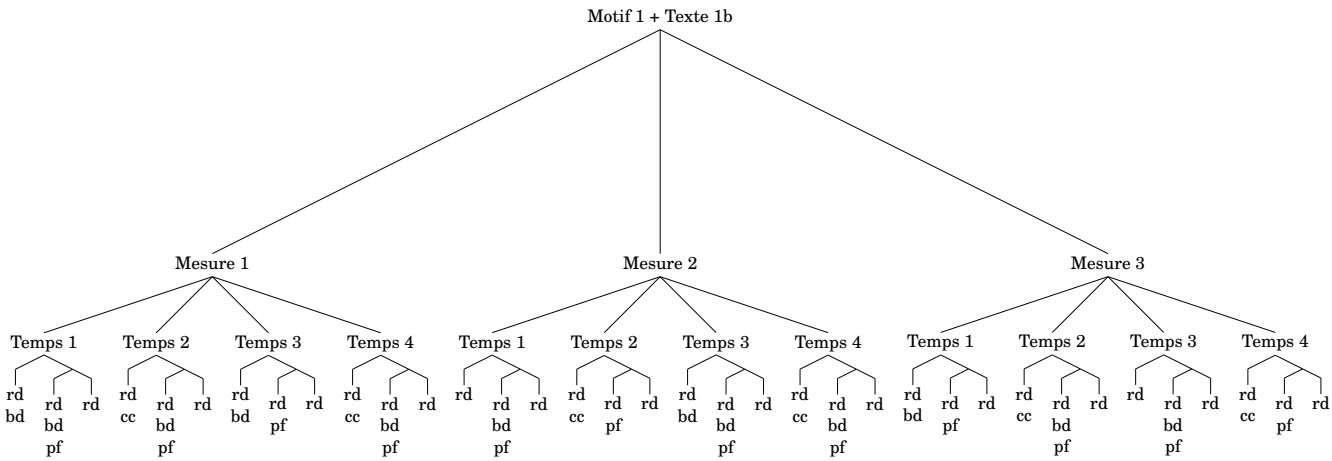
Motif 2, texte 2a.

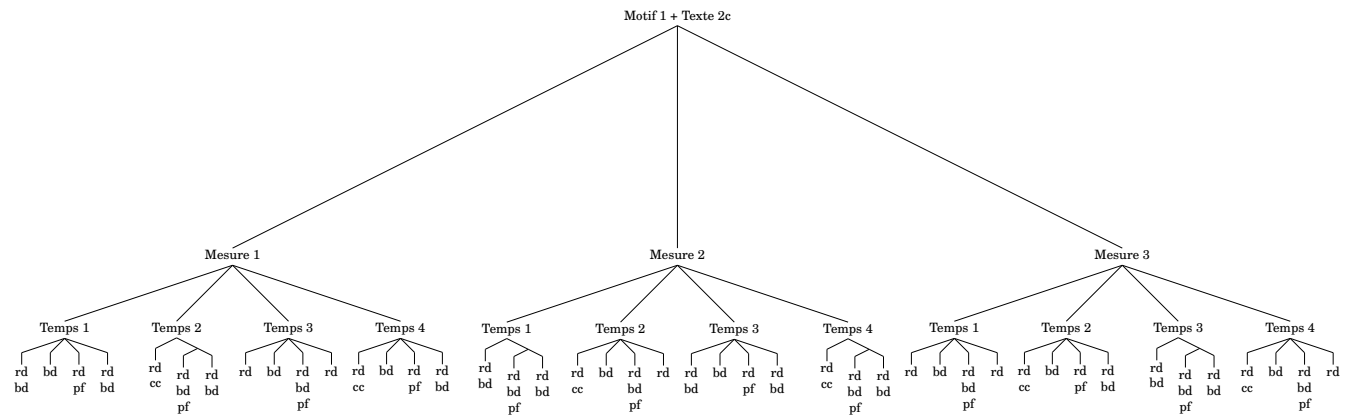
Motif 2, texte 2b.

Motif 2, texte 2c.

## Représentation des systèmes en arbres de rythmes



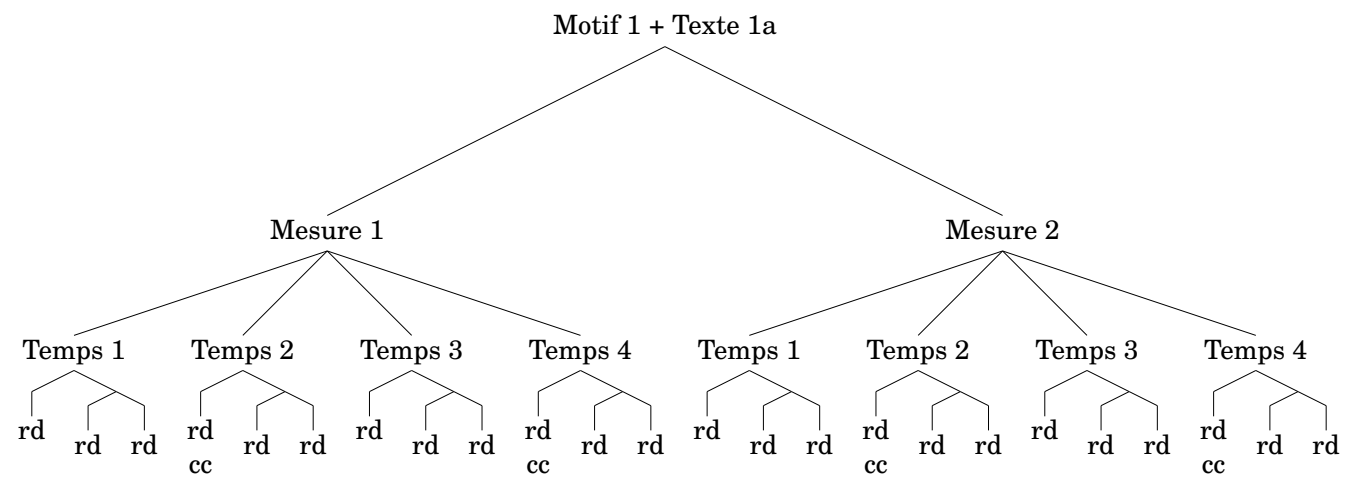




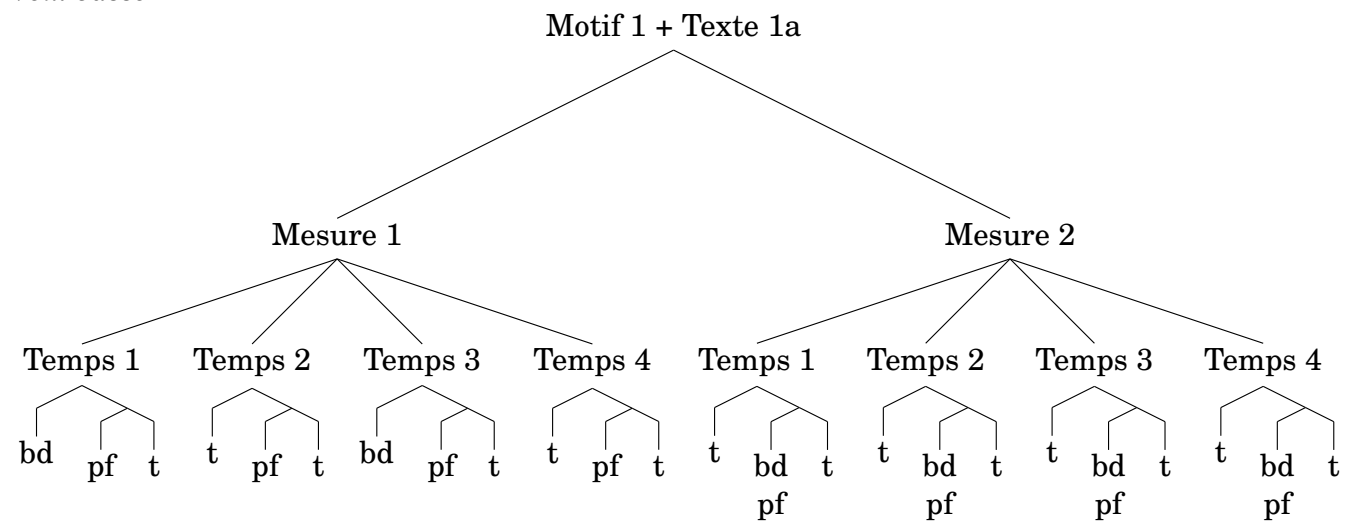
## Séparation des voix

## Motif 1 + Texte 1a

*Voix haute*

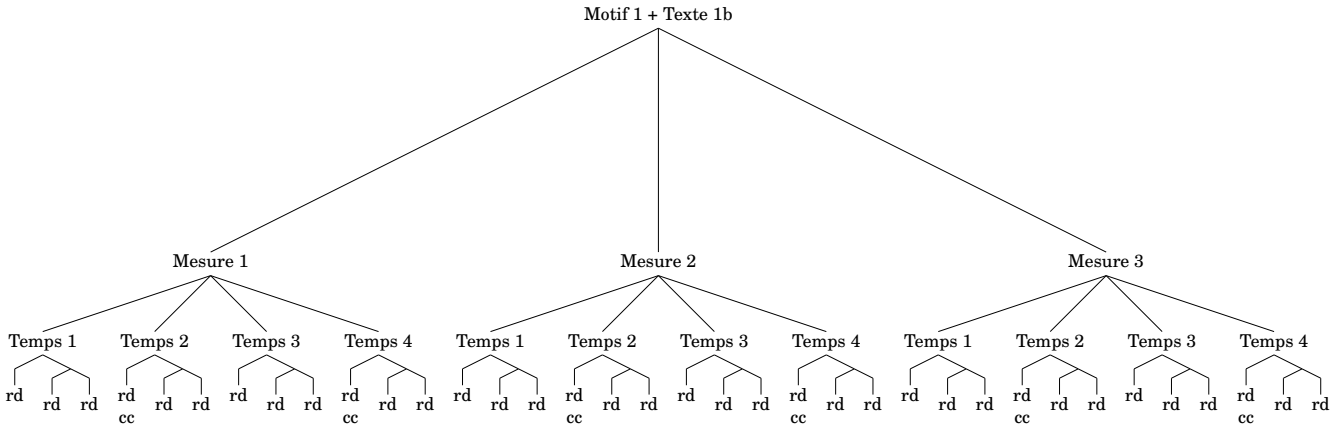


*Voix basse*

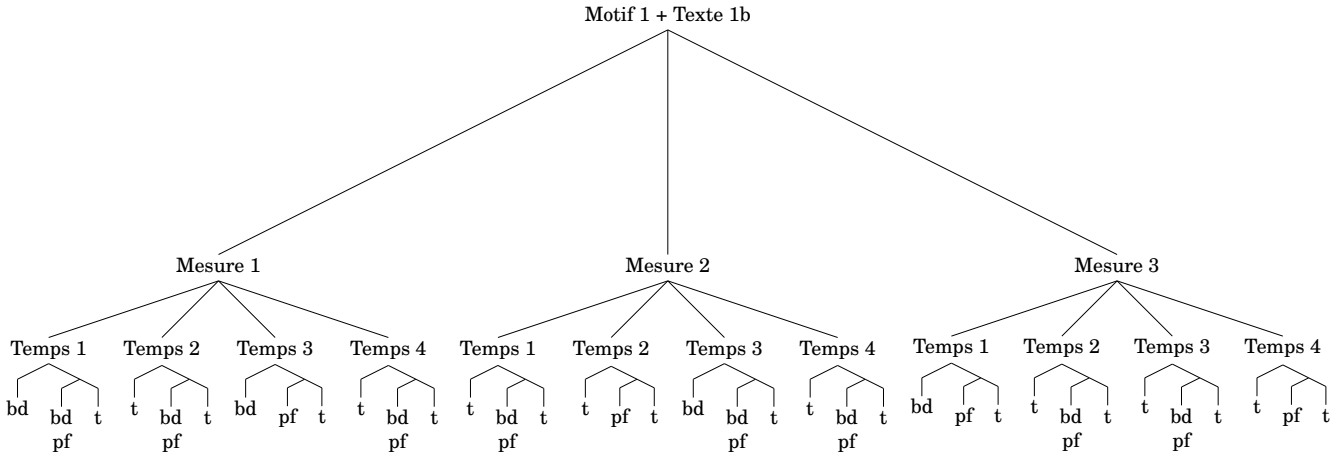


Motif 1 + Texte 1b

Voix haute

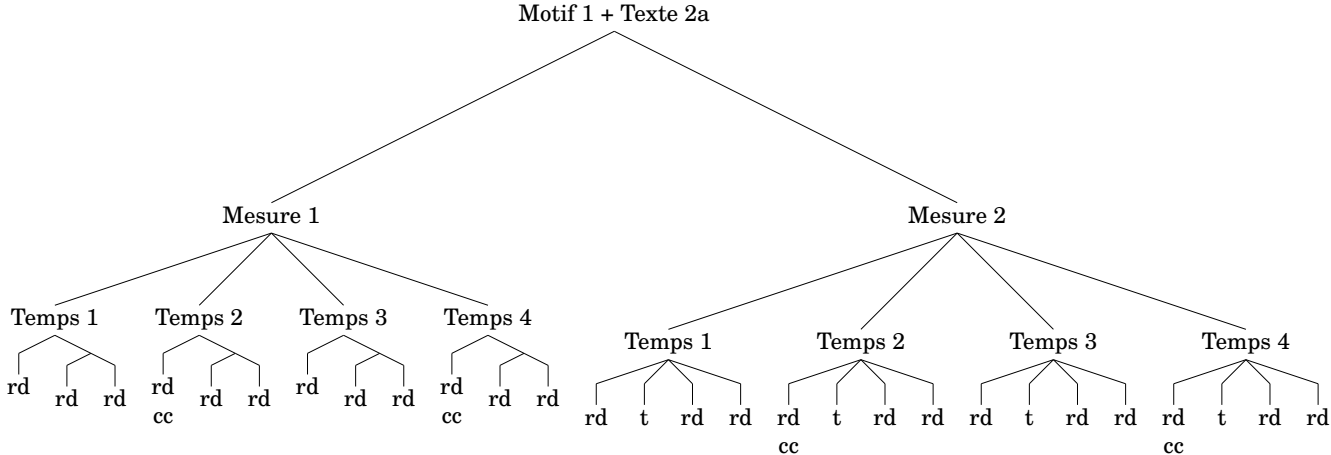


Voix basse

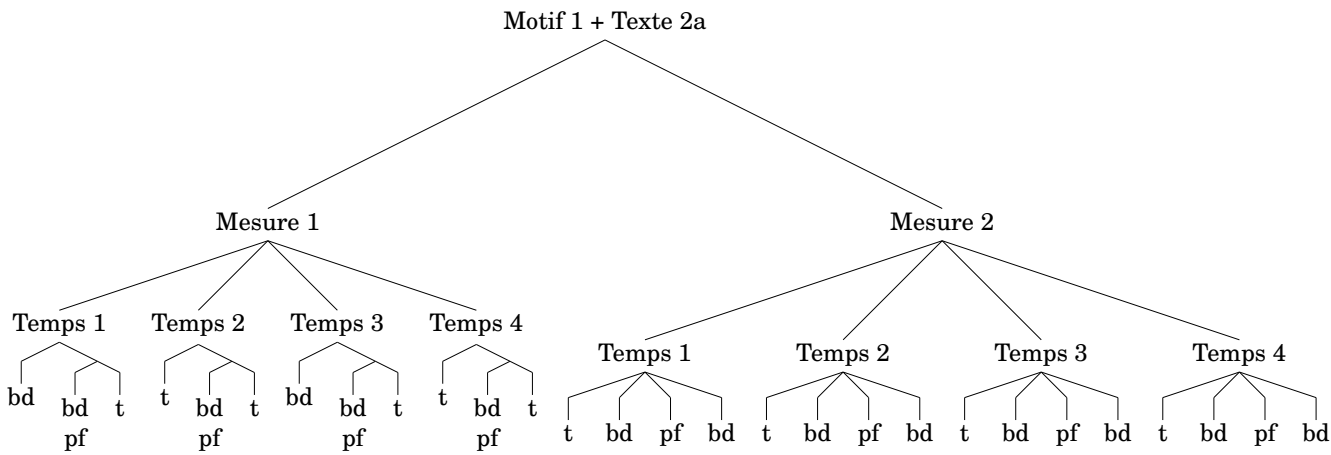


Motif 1 + Texte 2a

Voix haute

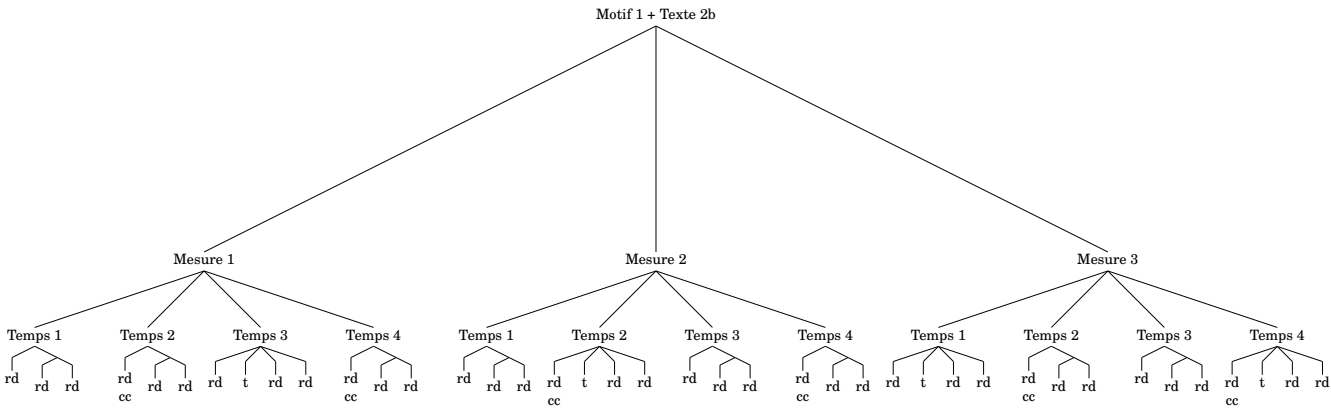


Voix basse

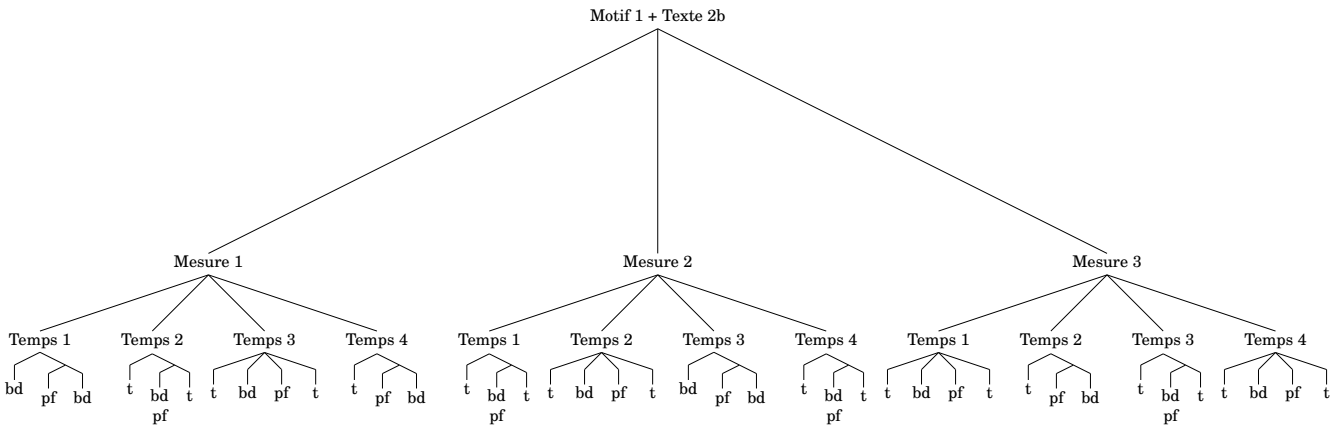


Motif 1 + Texte 2b

*Voix haute*

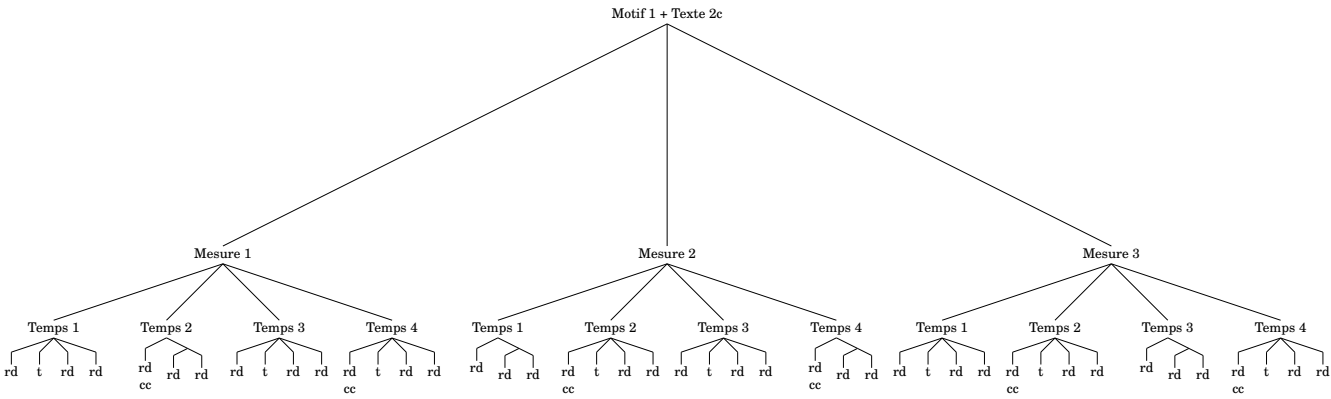


*Voix basse*

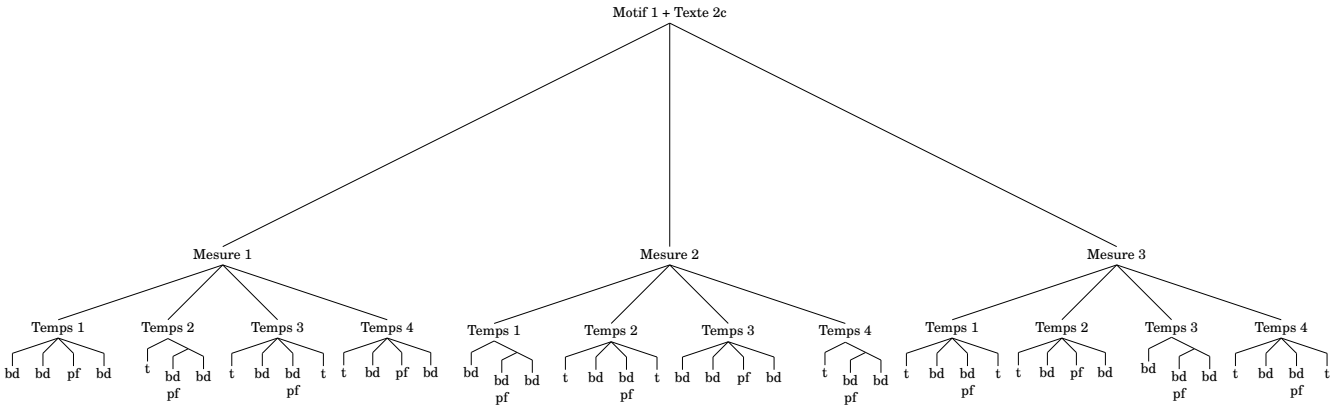


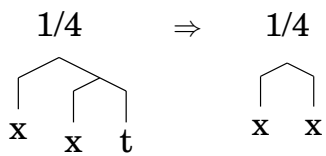
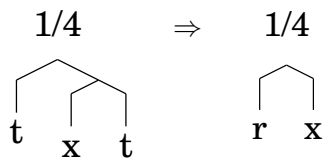
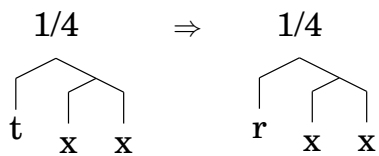
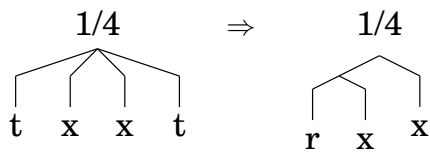
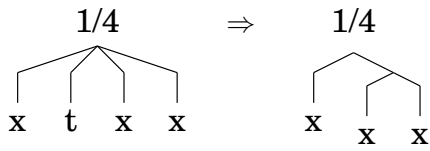
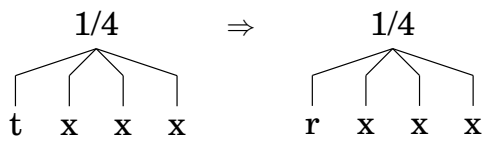
Motif 1 + Texte 2c

*Voix haute*



*Voix basse*



**Règles de réécriture pour le 4/4 binaire**

### 4.2.2 Expérience 2

#### Partition de référence pour l'ouput

The image displays a musical score for three staves, likely representing different voices or instruments. The notation includes various note values, slurs, and triplet markings (indicated by a '3' and a bracket). The first staff begins with a treble clef and a common time signature (C). The second staff begins with a bass clef. The third staff begins with a bass clef and a '5' marking. The score is enclosed in a rectangular frame.

*En cours...*



### 4.2.3 squant : parsing du fichier midi

squant lit le midi

grammaire wta qui détermine le poids

La distance est automatiquement déterminée par squant

distance à l'input

complexité de la notation

On veut minimiser le coût et la distance  $\Rightarrow$  Trouver un compromis.

`./build/squant2 -h`

Essayer le lire un fichier midi avec squant2 lire mesure par mesure Regarder les wta(grammaire)

#### Quelques tests de lecture midi

Les 4 messages suivants sont présents dans tous les tests qui suivent :

```
[ info] schema file: test/schema/schema-01.wta (??? weight model option)
[warning] no declaration MAX\_GRACE in grammar file test/schema/schema-01.wta
[warning] no declaration TIMESIG in grammar file test/schema/schema-01.wta
[warning] MIDIfile has not joined tracks
```

```
./build/squant2 -v 4 -a test/schema/schema-01.wta -m 004_jazz-funk_116_beat_4-
[error] at least one of the options -bars or -barsec mandatory
```

```
./build/squant2 -verbosity 4 -schema test/schema/schema-01.wta -midi 004\_jazz
```

```
squant2: /home/martin/qparselib/src/schemata/SymbLabel.cpp:44: static label_t
Abandon (core dumped)
```

Tester squant2 avec le fichiers midi du corpus du gitlab

La commande suivante :

```
build/squant2 -v 5 -a ./test/schema/schema-03-R.wta -m ~/corpus-master_qparselib
```

Donne :

- (1)  $3(\bullet, \bar{2}:2(\bullet, ), )$
- (2)  $3(\bullet, \bar{2}:2(\bullet, \bullet), )$
- (3)  $3(\bullet, \bar{2}:2(\bullet, \circ), )$

Pour comprendre les grammaires :

Regarder les fichiers wta commentés.

<https://qparse.gitlabpages.inria.fr/docs/scientific/>

A\_Parse-based\_Framework\_for\_Coupled\_RhythmQuantization\_and\_Score\_Structuring.pdf

Réfléchir au coût de notation (grace notes, etc.)

**cluster.md****Contributions**

Contribution sur la branch « distance » dans :

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

## **4.3 Conclusion**

Conclusion de ce chapitre.



## RÉSULTATS

### Sommaire

5.1	Introduction . . . . .	45
5.2	Contenu . . . . .	45
5.3	Conclusion . . . . .	45

*Résultats (chapitre 5) : les résultats obtenus sur chacune des expériences ;*

### 5.1 Introduction

Dans ce chapitre...

### 5.2 Contenu

Une section dans ce chapitre...

### 5.3 Conclusion

Conclusion de ce chapitre.



## DISCUSSION

### Sommaire

6.1	Introduction . . . . .	47
6.2	Contenu . . . . .	47
6.3	Conclusion . . . . .	47

*Discussion (chapitre 6) : la discussion des résultats obtenus (quelle expérience a produit les meilleurs résultats, de manière globale, dans le détail des catégories) avec, si possible, une analyse des erreurs pour comprendre les possibilités d'amélioration ;*

### 6.1 Introduction

Dans ce chapitre...

### 6.2 Contenu

Une section dans ce chapitre...

### 6.3 Conclusion

Conclusion de ce chapitre.





# CONCLUSION GÉNÉRALE

*Conclusion : la conclusion globale du mémoire.*

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



## BIBLIOGRAPHIE

- [1] 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 page 7.
- [2] 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 8, 13 et 14.
- [3] Emmanouil Benetos, Simon Dixon, Dimitrios Giannoulis, Holger Kirchhoff, and Anssi Klapuri. Automatic music transcription : Challenges and future directions. *Journal of Intelligent Information Systems*, 41, 12 2013. – Cité page 13.
- [4] Florent Jacquemard, Pierre Donat-Bouillud, and Jean Bresson. A Term Rewriting Based Structural Theory of Rhythm Notation. Research report, ANR-13-JS02-0004-01 - EFFICACe, March 2015. – Cité page 25.
- [5] Florent Jacquemard, Adrien Ycart, and Masahiko Sakai. Generating equivalent rhythmic notations based on rhythm tree languages. In *Third International Conference on Technologies for Music Notation and Representation (TENOR)*, Coruña, Spain, May 2017. Helena Lopez Palma and Mike Solomon. – Cité page 25.

