

LOG4199 - Projet de fin d'études
Transcription automatique de guitare

Martin Bisson (1208362)

5 décembre 2006

Sommaire

Le présent rapport contient les résultats du travail effectué dans le cadre du cours LOG4199 de l'École Polytechnique de Montréal, soit le projet de fin d'études en génie logiciel. Il traite du problème de transcription automatique de musique, spécialement de pièces interprétées à la guitare.

Le présent travail vise à explorer les solutions à une partie du problème de transcription automatique de musique, c'est-à-dire l'extraction d'une liste de notes, avec leur temps de début et leur durée, à partir d'un signal sonore numérique.

Pour ce faire, un prototype, basé sur une analyse de courts segments d'un morceau musical à l'aide de la transformée de Fourier, a été développé pour transcrire des pièces ne présentant que peu de difficulté. Ensuite, de nombreuses techniques d'analyse de signaux numériques ont été explorées afin d'améliorer le prototype pour le rendre capable de transcrire des pièces jouées à la guitare.

La version finale du logiciel est capable de transcrire, avec quelques notes oubliées et certaines notes identifiées faussement apparaissant avec l'augmentation de vitesse, une partie de Rondo Alla Turca, de Mozart, jouée à 5.333, 6.667 ou 8 notes par seconde.

Les objectifs de ce projet ont donc été atteints, soit d'arriver à obtenir un prototype fonctionnel et à explorer de nombreuses méthodes de traitement numérique de signaux. Certaines conclusions ont pu être dégagées autant en ce qui a trait à l'analyse de signaux numériques en général qu'au problème de transcription automatique de musique en particulier.

Table des matières

1	Introduction	1
1.1	Problème abordé	1
1.2	Intérêt	2
1.3	Plan du travail	3
2	Problématique	4
2.1	Description du problème	4
2.1.1	Signal numérique vers ensemble de notes	4
2.1.2	Ensemble de notes vers partition	5
2.1.3	Réalité	6
2.2	Tour de lecture	7
2.3	Portée du travail	9
3	Méthodologie	11
3.1	Étapes du développement	11
3.1.1	Tour de lecture	11
3.1.2	Environnement de travail	11
3.1.3	Prototype	13
3.1.4	Améliorations	13
3.1.5	Rédaction du rapport	14
3.1.6	Période tampon	14
3.2	Échéancier	15
4	Résultats	16
4.1	Premiers pas : simplement Fourier	16
4.1.1	Explications	16
4.1.2	Objectifs	18
4.1.3	Détails d'implantation	18
4.1.4	Résultats	20
4.2	Échelle logarithmique	23
4.2.1	Explications	23
4.2.2	Objectifs	24
4.2.3	Détails d'implantation	24
4.2.4	Résultats	25
4.3	Guitare enregistrée	27
4.3.1	Explications	27

4.3.2	Objectifs	27
4.3.3	Détails d'implantation	27
4.3.4	Résultats	28
4.4	Ajout de zéros	31
4.4.1	Explications	31
4.4.2	Objectifs	31
4.4.3	Détails d'implantation	32
4.4.4	Résultats	32
4.5	Utilisation de fenêtres non rectangulaires	33
4.5.1	Explications	34
4.5.2	Objectifs	35
4.5.3	Détails d'implantation	35
4.5.4	Résultats	35
4.6	Autres...	36
5	Discussion	47
5.1	Autres outils de transcription automatique	47
5.1.1	SONIC	47
5.1.2	AmazingMIDI	49
5.1.3	Autres	50
5.2	Conclusions générales	51
5.2.1	Nécessité d'abstraction	51
5.2.2	Résolution temporelle/fréquentielle	52
5.2.3	Bruit	52
5.2.4	Techniques	53
5.3	Conclusions relatives à la TAM	53
5.3.1	Exponentialité de la croissance des fréquences	53
5.3.2	Importance de l'interprétation	54
5.3.3	Utopie d'une solution parfaite	55
5.4	Poursuite des travaux	56
6	Conclusion	58
7	Bibliographie	59
8	Glossaire	62
9	Annexe A : Description du projet	64

10 Annexe B : Décalage temporel	66
11 Annexe C : CSound	67
12 Annexe D : Effet d'un décalage des valeurs d'un signal	68
13 Annexe E : Rondo Alla Turca	71

Remerciements

Je tiens à remercier le professeur François-Raymond Boyer, le directeur de ce projet de fin d'études, d'avoir accepté de partager son expertise et d'avoir été aussi généreux de son temps, pour ce travail et en d'autres circonstances.

Liste des tableaux

1	Échéancier	15
2	Notes du son généré comme entrée du premier système	19
3	Notes transcrites par le premier système	19
4	Notes du son généré comme entrée du premier système modifié	24
5	Notes transcrites par le premier système modifié	25

Table des figures

1	Conversion d'une représentation à l'autre	7
2	Première transcription à 4096 échantillons par segment	21
3	Première transcription à 8192 échantillons par segment	22
4	Transcription à l'aide de représentation fréquentielle logarithmique	26
5	Transcription d'une pièce avec beaucoup de bruit	29
6	Transcription logarithmique d'une pièce avec beaucoup de bruit	30
7	Ajout de zéros (facteur 1)	38
8	Ajout de zéros (facteur 2)	39
9	Ajout de zéros (facteur 4)	40
10	Ajout de zéros (facteur 8)	41
11	Ajout de zéros (facteur 16)	42
12	Ajout de zéros (facteur 64)	43
13	Problème d'harmoniques	44
14	Signal à 8 Hz échantillonné à 100 Hz pendant 1 seconde	45
15	Signal à 8.5 Hz échantillonné à 100 Hz pendant 1 seconde	45
16	Signal à 8.5 Hz avec fenêtre de Hamming	46
17	Résultat de SONIC pour Au Clair de la Lune	48
18	Résultat de SONIC pour Rondo Alla Turca	48
19	Capture d'écran de AmazingMIDI	50
20	Au Claire de la Lune	66
21	Au Claire de la Lune avec un problème de décalage	66
22	Transcription d'une pièce avec décalage	69
23	Transcription logarithmique d'une pièce avec décalage	70
24	Rondo Alla Turca	71

Liste des abréviations

DFT *Discrete Fourier Transform*, ou transformée de Fourier discrète.

FFT *Fast Fourier Transform*, ou transformée rapide de Fourier.

MIDI *Musical Instrument Digital Interface*.

TAM Transcription automatique de musique.

1 Introduction

L'informatique prend une place de plus en plus importante dans pratiquement tous les champs d'activités humaines : les mathématiques, la comptabilité, le divertissement et même la médecine ne sont que des exemples de domaines qui ont été grandement influencés par le développement d'applications informatiques au cours des dernières décennies.

Par contre, il peut être extrêmement difficile de développer une solution automatisée à certains problèmes, malgré que ces derniers peuvent s'avérer relativement faciles à résoudre par l'humain. On pense entre autres à la reconnaissance visuelle de toute sorte (formes, écriture, perception spatiale), à l'apprentissage et l'adaptation, etc. En effet, un être humain peut relativement facilement évaluer la distance à laquelle se trouve un objet dans une pièce ou déchiffrer une écriture manuscrite de mauvaise qualité, alors que le développement de logiciels capables des mêmes habiletés est extrêmement complexe.

Une des raisons principales de la difficulté de la résolution d'un tel problème est que l'humain, parfois même sans s'en rendre compte, utilise une panoplie de « connaissances générales », en apparence sans rapport avec le problème, pour le résoudre plus facilement. Par exemple, un être humain qui tente de lire une lettre manuscrite se servira de son vocabulaire ainsi que sa connaissance du langage pour tenter de reconnaître les lettres difficiles à lire. Il pourra également se servir de la signification des mots pour en tirer un sens qui lui permet d'éventuellement remplacer les parties manquantes ou qu'il ne peut reconnaître simplement en tentant de lire caractère par caractère.

1.1 Problème abordé

Le problème abordé dans le cadre de ce projet de fin d'études est un problème de ce type : la transcription automatique de musique (TAM)¹. On appelle *transcription de musique* le travail d'un musicien qui, souvent suite à une écoute répétée et un travail itératif, reconstruit la partition, sous une forme ou sous une autre, représentant un ou plusieurs instruments d'une cer-

¹Les termes plus spécifiques au domaine étudié sont définis dans le glossaire.

taine performance musicale.

Cette tâche peut être accomplie de plusieurs façons, mais elle exige souvent une certaine connaissance de la musique en général, de la théorie musicale, du style de musique qu'on cherche à transcrire, etc., en plus d'une bonne dose d'« intuition musicale », une certaine expérience, des habilités artistiques, une capacité à faire des liens entre différentes parties ainsi qu'une capacité à synthétiser des informations provenant de différents indices pour arriver à un résultat. Ces exigences correspondent à des forces du cerveau humain, ce qui fait de la transcription de la musique un problème multidisciplinaire très complexe à automatiser.

1.2 Intérêt

Bien que souvent la seule complexité d'un problème soit suffisante pour susciter l'intérêt des chercheurs à tenter de le résoudre, l'automatisation de la transcription de musique pourra fournir des outils très intéressants.

Les premiers à en bénéficier seraient sans doute les musiciens, autant les amateurs que les professionnels. En effet, l'écriture d'une partition pourrait être grandement facilitée par un système de transcription automatique. Il est déjà possible, par exemple à l'aide d'un clavier MIDI², d'écrire une partition en la jouant. Or, cela n'est pas possible pour tous les instruments, notamment la plupart des instruments à vent et à cordes. La TAM permettrait de généraliser cette technique d'écriture de partitions.

De plus, les musiciens amateurs cherchant à apprendre à jouer leur pièce préférée dont les partitions ne sont pas disponibles verraient leur tâche grandement simplifiée par la disponibilité d'outils de TAM. Ceux-ci les guideraient dans la tâche d'identification des notes à jouer.

Par ailleurs, ils existe d'autres applications potentielles de la TAM un peu plus secondaires. On pense entre autres à la possibilité de développer des logiciels d'apprentissage d'instruments, ou même des jeux, capables de reconnaître les fausses notes, le mauvais rythme, etc. Il existe également des

²MIDI pour *Musical Instrument Digital Interface*.

systèmes dits de *recherche par fredonnement*³ qui permettent de rechercher le nom et l'interprète d'une chanson en la fredonnant([4]).

1.3 Plan du travail

Le présent rapport contient les détails du travail qui a été effectué, structuré de la manière suivante :

- Le chapitre 2 décrira plus en détails le problème de la transcription automatique. On y retrouvera également un survol des travaux accomplis dans ce domaine ainsi qu'une explication de la portée de ce travail.
- Le chapitre 3 définira l'approche qui a été utilisée au cours de ce travail et proposera un échéancier.
- Le chapitre 4 décrira chacune des différentes techniques employées. Il décrira également, pour chacune d'elles, le fondement théorique derrière la technique, les objectifs de son utilisation, les détails ou problèmes d'implantation s'il y a lieu pour finalement donner les résultats de l'emploi de cette technique.
- Le chapitre 5 synthétisera les résultats obtenus et les comparera à différents outils que l'on peut qualifier de « produits finis » de TAM.

En somme, après cette brève introduction du sujet et l'élaboration du plan du travail, on peut maintenant passer à une définition plus élaborée du problème traité, ce qui sera fait dans le chapitre suivant.

³Traduction libre de « Search By Humming ».

2 Problématique

Ce chapitre donnera une description un peu plus approfondie du problème de la transcription automatique de musique, des travaux déjà effectués sur le sujet ainsi que la portée de ce travail.

2.1 Description du problème

Tel que mentionné en introduction, la TAM est un problème qui est difficile à automatiser, entre autres parce qu'il requiert un certain niveau d'« intuition » ou d'« instinct » musical ainsi que la synthèse et la compréhension d'une multitude de sources potentielles d'information.

Afin d'arriver à aborder un problème aussi complexe, l'auteur divise le problème de TAM en deux sous-problèmes : la conversion du signal numérique en un ensemble de notes, puis la conversion de l'ensemble de notes en une partition.

2.1.1 Signal numérique vers ensemble de notes

L'entrée d'un système de TAM est un simple signal sonore numérique, sous la forme d'un ensemble de nombres en virgule flottante. Chacun de ces nombres représente l'intensité du signal à un temps donné. Il est extrêmement difficile de tirer de l'information utile pour la TAM à partir de cette représentation. En fait, pratiquement la seule information qui peut être tirée de cette représentation a trait au rythme, car les maximums d'intensité correspondent généralement à des notes jouées.

Toutefois, cette représentation d'un signal ne permet pas d'obtenir de l'information sur la hauteur des notes jouées. Il existe une autre manière de représenter un signal qui s'avère très intéressante pour la TAM. En effet, il est possible de représenter tout signal numérique comme la somme de sinusoides de différentes fréquences. Comme la hauteur perçue d'une note (ce qui distingue un Do d'un Ré par exemple) est directement reliée à sa fréquence, cette représentation est très prometteuse pour la TAM. La façon de passer d'une représentation à l'autre sera abordée plus loin.

Les principaux défis de cette étape d'extraction d'un ensemble de notes sont :

- Identifier la bonne hauteur de la note, peu importe si elle est très grave ou très aigüe.
- Identifier avec précision le début et la fin des notes.
- Reconnaître plusieurs notes jouées en même temps.
- Distinguer plusieurs notes jouées en même temps des harmoniques d'une même note.
- Reconnaître des effets particuliers à un instrument, comme le « bend » à la guitare.

Ainsi, on peut donc tenter de résoudre la première partie du problème, c'est-à-dire chercher, à partir du signal numérique de départ, à obtenir la liste des notes qui composent la mélodie transcrite. Après cette étape, on aura donc un ensemble de notes pour chacune desquelles on a le moment de départ, la durée, puis éventuellement d'autres informations comme l'intensité ou l'application d'un effet quelconque. Il est donc possible, à partir de cette liste de notes, de « rejouer » la mélodie transcrite. Si cette partie de la transcription s'est bien déroulée, la mélodie reproduite devrait être en tout point semblable à la mélodie originale. Cette liste constitue déjà de l'information très précieuse pour un musicien cherchant à transcrire une mélodie. Toutefois, il reste encore beaucoup de travail pour arriver à obtenir une véritable partition.

2.1.2 Ensemble de notes vers partition

En effet, bien que la liste de notes constitue un bon départ et permette de rejouer la mélodie, on est encore bien loin d'avoir une partition musicale. Il reste une seconde étape qui consiste à extraire une certaine structure de l'ensemble de notes pour arriver à une partition.

Les feuilles de musique représentent une mélodie d'une manière très structurée. On n'y retrouve pas que la durée et la hauteur de chaque note, mais également la notion de mesures et de temps (ainsi que les fractions et les multiples de temps). Comme la musique est une forme d'art, cette structure n'a pas nécessairement la rigidité qui faciliterait l'organisation de la liste de notes, car le résultat d'une pièce musicale suit bien souvent les fantaisies de

l'interprétation du musicien. Ainsi, de nombreux cas rendent l'extraction de la partition plus difficile :

- Le rythme change (accélère ou décélère) au cours de la pièce.
- Une brève variation de structure se produit : le musicien tient une note un quart de temps de plus qu'un métronome lui aurait permis, ce qui fait que le système croit que toutes les notes suivantes sont décalées d'un quart de temps, ce qui rend la partition difficile à lire et cette dernière ne reflète pas la véritable structure de la pièce. L'annexe B montre un exemple de ce décalage.
- La multitude d'interprétations possibles peut causer des problèmes : interpréter un rythme comme étant du 3/4, du 4/4, du 6/8 ou même du 7/4⁴ changera complètement l'allure de la partition produite.

Il n'est donc pas trivial de passer d'une liste de notes à une partition. Cette seconde étape constitue une autre partie complexe du problème de TAM.

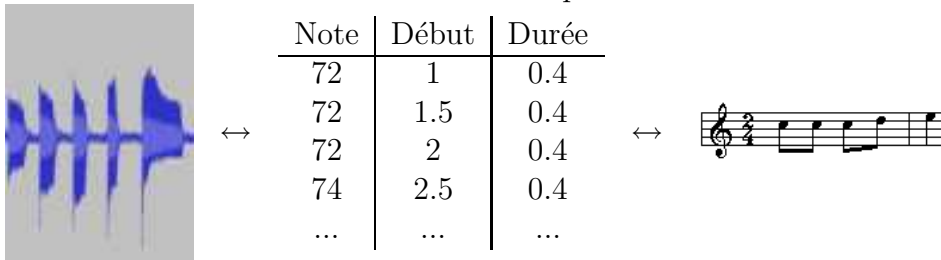
2.1.3 Réalité

Bien qu'il puisse sembler logique de procéder d'abord à la conversion du signal en une liste de notes et de durées, puis de convertir cette liste en une partition, cette approche ne correspond pas du tout à la façon dont un musicien (humain) travaille. Effectivement, l'être humain procéderait à ces deux étapes d'une manière non pas séquentielle, mais plutôt parallèle. L'identification des notes et de la structure n'est donc pas séparée en tâches distinctes : le résultat de l'une donne de l'information utile à l'autre. De plus, l'être humain utilise une panoplie d'autres sources d'information, telles que la connaissance de l'artiste ou du style de musique de la pièce à transcrire.

Ce type de synthèse que le cerveau humain semble en mesure de faire relativement facilement constitue un modèle de solution extrêmement complexe pour un système informatique. Aborder la TAM avec cette approche

⁴Cette notation sert à décrire la composition d'une mesure. Le premier nombre indique le nombre de notes ou de temps par mesure, tandis que le second nombre indique la durée de ce temps : 1 correspond à une ronde (4 temps), 2 à une blanche (2 temps), 4 à une noire (1 temps), 8 à une croche (1/2 temps), etc. Ainsi, 3/4 signifie que chaque mesure contiendra 3 (3/4) noires (3/4), ou l'équivalent en durée.

FIG. 1 – Conversion d’une représentation à l’autre



plutôt basée sur l’intelligence artificielle déborde du cadre de ce travail, tel qu’il le sera décrit plus loin dans ce chapitre. Toutefois, la manière qu’ont les musiciens de procéder à la TAM constitue une piste intéressante.

Finalement, la figure 1 montre le passage d’une représentation à l’autre. À noter qu’il est beaucoup plus facile de passer d’une représentation à l’autre en allant de droite à gauche plutôt que de gauche à droite. Pour faire une analogie informatique, le passage de droite à gauche est comparable à la conversion d’un fichier de code source en un exécutable (facilement automatisable, il s’agit du travail du compilateur, de l’assembleur et de l’éditeur de liens) alors que le passage de gauche à droite se compare à la conversion d’un exécutable à un fichier de code source, suite à un désassemblage (beaucoup plus difficilement automatisable, car des informations sont manquantes et doivent être déduites).

2.2 Tour de lecture

Le problème de transcription automatique a déjà été étudié par des chercheurs universitaires. Il serait donc pertinent, pour bien situer ce projet de fin d’études, de se référer à ces travaux pour en apprécier les différentes approches.

Certains auteurs se sont attaqués à la transcription automatique de mélodies chantées. Bien que monophonique, la voix est un « instrument de musique » extrêmement expressif, ce qui rend donc sa transcription plutôt complexe. Weihs et Ligges ([24]) ont développé une technique basée sur la séparation de la pièce en petits segments qui sont ensuite classifiés selon la hauteur de la note la plus forte qu’ils contiennent. Leur heuristique utilise

certaines informations spécifiques à la voix, telle que l'étendue du registre des notes atteignables par le type de voix (soprano, tenor, etc.) en action ou l'étendue des notes que l'on prévoit avoir dans la pièce transcrite. De plus, on utilise un aplannisseur médian⁵ pour ne pas considérer un vibrato comme une trille. Même si la transcription de la voix diffère de celle de la guitare, l'utilisation d'information spécifique à l'instrument peut s'avérer utile. L'approche de la gestion du vibrato constitue une première approximation tolérable, mais n'est toutefois pas une solution complète puisqu'on voudrait plutôt être en mesure de reconnaître cet effet plutôt que simplement le masquer.

Weihls et Ligges ([25]) poursuivent leur travail ([24]) en tentant d'optimiser les différents paramètres de leur heuristique, tels que les seuils d'intensité limite utilisés dans la classification ou la taille de l'aplannisseur médian, en fonction de l'interprète de la pièce à transcrire. Il est évident que leurs résultats ne s'appliquent pas directement à la transcription de la guitare, mais le concept d'ajustement des paramètres de la transcription en fonction de l'instrument transcrit est certes intéressant. Toutefois, leur approche nécessite un certain entraînement du système, ce qui ajoute une contrainte non souhaitable à l'utilisation du système qui sera développé dans le cadre de ce projet de fin d'études.

Klapuri est un chercheur très actif dans le domaine de la transcription automatique. Il s'est attaqué à plusieurs facettes du problème, notamment l'identification de fréquences multiples dans un signal ([10] et [11]), inévitable pour la transcription de pièces polyphoniques. Son approche est basée sur les travaux de Tolonen et Karjalainen ([23]) et ceux de De Cheveigné et Kawahara ([5]). Ces derniers proposent une méthode itérative d'identification de fréquences multiples où une première fréquence est identifiée, puis retirée du signal original avant de tenter de trouver les autres fréquences. Suite à une première approximation des fréquences, on peut ensuite itérer pour estimer avec plus de précision chacune des fréquences en ayant soustrait les autres du signal.

Klapuri compare également la transcription automatique de musique à la reconnaissance vocale au niveau du bénéfice possible d'information spécifique au domaine pour l'orientation de la solution. Sans entrer dans les détails, il a

⁵Traduction libre de *median smoother*.

conduit une étude statistique ([10]) sur 359 pièces en format MIDI pour identifier les combinaisons de notes les plus fréquentes pour en conclure qu'avec les notes seules, les combinaisons les plus fréquentes sont les accords mineurs et majeurs. Ainsi, sans connaissances musicales ni paramètres à ajuster pour une heuristique quelconque, un système pourrait ainsi apprendre de la théorie musicale qui l'aiderait à faire des choix plus éclairés ou du moins mieux guidés lors de la transcription automatique. Ce projet de fin d'études ne s'attaquera probablement pas au problème de la transcription polyphonique, mais il s'agit là de pistes intéressantes.

Finalement, d'autres auteurs, comme Temperley ([22]), ont tenté, à partir d'une liste de notes, d'automatiser le processus d'extraction de structure pour la production d'une partition. Le présent projet ne s'occupera pas de cette partie de la transcription, car le travail sera concentré sur l'extraction des notes, de leur début et de leur fin, plutôt que la conversion de ces notes en une partition.

2.3 Portée du travail

Ce travail de recherche sur la TAM ne touchera qu'à la première étape (section 2.1.1) du processus décrit plus haut. On s'intéressera donc davantage aux aspects du problème relatifs au traitement de signaux numériques plutôt qu'à ceux qui relèvent de l'intelligence artificielle. Ainsi, le code développé dans le cadre de ce projet visera à obtenir, à partir d'un signal numérique représentant une pièce jouée à la guitare, la liste des notes qui composent une pièce musicale, avec leur durée.

Selon les normes de l'École Polytechnique, le projet de fin d'études est un travail de 135 heures, ce qui est relativement peu pour aborder un problème aussi complexe que la TAM. Les ambitions et objectifs de ce projet seront donc corrélés à ce nombre d'heures allouées.

Tout d'abord, il est important de préciser que ce projet est d'abord et avant tout exploratoire. L'objectif principal de sa réalisation n'est pas d'arriver à une solution parfaite, mais bien d'explorer le problème de la TAM. On cherche donc à attaquer le problème, à se familiariser avec les difficultés communes de l'analyse de signaux numériques (effet du bruit, compromis résolution temporelle/fréquentielle, etc.) et à découvrir certaines pistes de

solutions. Cet objectif influencera beaucoup la méthodologie utilisée pour le développement de ce projet (voir chapitre 3).

Le but de ce travail *n'est pas* d'arriver à une solution nouvelle ou de réinventer la science. Il n'est pas non plus d'arriver à un résultat qui se compare avantageusement aux différentes solutions, commerciales ou non, disponibles sur le marché ou sur Internet. Tel que mentionné plus haut, il serait irréaliste d'espérer arriver à de tels résultats en 135 heures, d'où le choix d'objectifs plus modestes. Finalement, le logiciel développé sera très loin d'un produit fini. *Aucune* interface ne sera développée, le travail sera toujours fait en modifiant directement le code. Encore une fois, ce choix est guidé par les objectifs choisis pour ce projet, qui sont d'explorer les problèmes et solutions sans pour autant arriver avec un produit fini, robuste, performant, commercialisable, etc.

La méthodologie proposée pour atteindre ces objectifs sera décrite dans le chapitre suivant.

3 Méthodologie

Comme ce projet de fin d'études est d'abord et avant tout un travail exploratoire, les processus classiques de développement logiciel, par exemple la fameuse suite « analyse → conception → implantation → tests », ne sont pas vraiment applicables. Une méthodologie personnalisée sera donc proposée : les différentes étapes seront d'abord décrites brièvement⁶ et un échéancier sera ensuite proposé.

3.1 Étapes du développement

Les différentes étapes proposées seront ici énumérées.

3.1.1 Tour de lecture

La première étape de ce travail consistera à prendre connaissance des différents travaux sur le sujet. Cette recherche permettra de se familiariser avec le sujet, de ne pas tomber dans les pièges que d'autres n'ont pu éviter et de fournir des pistes pour régler d'éventuels problèmes rencontrés.

Comme l'a dit Newton : « Si j'ai pu voir un peu plus loin que d'autres, c'est que je me suis hissé sur les épaules de géants. »

3.1.2 Environnement de travail

De nombreux outils seront nécessaires pour la réalisation du travail proposé. En plus des logiciels de développement habituels⁷ et des différents logiciels audio (enregistrer un son, le visualier, l'entendre, etc.), un certain cadre de travail⁸ a été développé.

Voici une brève liste des points de base qui ont été réglés avant de commencer le véritable développement :

- Lecture du son : comme le programme devra analyser un signal sonore numérique, il devra être en mesure de lire des fichiers de son standards,

⁶Les détails de chaque étape seront donnés au chapitre 4.

⁷Le développement se fera sous Linux ; on pense donc à `gcc`, `gdb`, `emacs`, etc.

⁸Traduction libre de *framework*

par exemple le format `.mp3` (qui ne sera probablement pas utilisé à cause des pertes potentielles dues à la compression) ou `.wav` (qui sera probablement choisi). Le but du projet n'était pas d'écrire un lecteur de fichiers audio, on utilisera plutôt une des bibliothèques *Open Source*, comme `libsndfile` ([8]) ou `SndObj` ([13]). À cause de sa simplicité d'utilisation, `libsndfile` a été retenue.

- Traitement du son : certaines classes ont été développées pour pouvoir facilement charger un fichier audio dans un objet qui permet un accès simple aux échantillons du son. Ces classes permettent également certaines opérations, comme l'affichage d'une représentation graphique du signal numérique, la copie d'une partie du signal, etc.
- Génération du son : les premières transcriptions automatiques seront effectuées sur des échantillons générés afin de contrôler plus facilement la nature de l'échantillon pour fournir un cas simple de transcription (son purement sinusoïdal, absence de bruit provenant de l'enregistrement, etc.). Plusieurs approches sont possibles : génération du son directement dans le code, utilisation de fichiers MIDI, utilisation de CSound⁹, etc.
- Calcul de la transformée de Fourier : le genre de traitement qui sera effectué dans le cadre de ce travail nécessitera sans aucun doute l'utilisation de la transformée de Fourier et de sa version extrêmement rapide, la transformée de Fourier rapide ou FFT. On a donc choisi d'utiliser une implantation reconnue pour être extrêmement efficace, soit `fftw` ([9]).
- Sortie des résultats : la transcription automatique développée dans le cadre de ce projet aura comme sortie une liste de notes avec leur durée. On devra toutefois être en mesure de juger de la qualité de la transcription. Il existe donc plusieurs formats de sortie ayant chacun leurs avantages. Le plus évident est sans doute une simple liste en format texte ; ce dernier fait toutefois de l'analyse des résultats une tâche assez fastidieuse. Il serait donc avantageux d'utiliser un format qui permettrait de facilement rejouer la liste de notes produite. Un des formats les plus connus qui permet d'y arriver est le format MIDI, qui a toutefois l'inconvénient de ne pas être facilement lisible sans logiciel MIDI. Le format CSound a donc été choisi.

⁹Voir l'annexe C pour une explication plus détaillée de CSound.

On voit donc qu'il y a une certaine quantité de travail préparatoire à faire afin d'obtenir un cadre de travail qui favorisera le développement du reste du projet.

3.1.3 Prototype

L'étape suivante sera de réussir à obtenir une première version fonctionnelle. Cette dernière devra être en mesure d'effectuer la transcription automatique de pièces ne présentant absolument aucune difficulté, c'est-à-dire :

- La pièce générée par ordinateur plutôt qu'enregistrée : il n'y a donc pas de bruit.
- L'« instrument » simulé joue ses notes à l'aide d'une onde sinusoïdale simple et pure, le signal sonore le plus simple qui soit.
- Chaque note est séparée de la précédente et de la suivante par un silence.
- La pièce est monophonique.
- Le tempo de la pièce est lent.

Tel qu'il le sera décrit dans le chapitre 4, la réalisation de ce prototype devrait pouvoir être faite seulement à l'aide des idées de l'auteur (qu'il n'est pas le premier à avoir bien sûr). Ce type de problème étant un cas simple de transcription automatique, les expérimentations de l'auteur devraient suffire à produire cette première version.

La dernière étape pour considérer le prototype initial comme complet sera de transcrire un son tout aussi simple, mais enregistré plutôt que généré. Un simple enregistrement sans trop de bruit et quelques ajustements de paramètres devraient permettre d'atteindre cet objectif à partir du premier prototype.

3.1.4 Améliorations

Une fois tout ce travail fait, on cherchera ensuite à améliorer ce qui aura été accompli. Des cas plus complexes, par exemple des pièces jouées rapidement, seront soumis au système jusqu'à l'atteinte d'un cas problématique. Des techniques seront ensuite appliquées pour tenter d'améliorer le système pour qu'il soit en mesure de transcrire ce morceau. On retentera l'expérience

avec des morceaux plus complexes, en répétant cette itération jusqu'à l'obtention d'un résultat satisfaisant, ou plus probablement jusqu'à la fin du temps disponible pour la réalisation de ce projet.

Cette méthodologie itérative permettra de s'attaquer à différents problèmes et d'évaluer différentes pistes de solutions, le tout dans le temps disponible, afin d'atteindre les objectifs de ce projet de fin d'études.

Les résultats seront présentés dans le chapitre 4. Chacune des techniques ou solutions employées sera évaluée sur différents aspects, par exemple :

- La résolution temporelle
- La résolution fréquentielle
- La performance (possibilité d'utiliser la FFT pour cette solution, nécessité de prétraitement, etc.)
- Les autres avantages (polyphonie, etc.)

3.1.5 Rédaction du rapport

Évidemment, une certaine plage de temps doit être réservée pour la rédaction du rapport. Pour un travail d'environ 30 pages, il n'est pas pessimiste d'estimer environ une heure par page. Bien qu'en général l'auteur écrit un peu moins de deux pages à l'heure, il faut tenir compte du temps de relecture, de pages qui seront plus longues à écrire étant donné la recherche de figures ou la construction de tables, etc. De plus, comme le rapport risque fortement de dépasser 30 pages, les 30 heures allouées ne seront pas de trop.

3.1.6 Période tampon

L'expérience personnelle de l'auteur ainsi que celle des développeurs qu'il a côtoyés révèle que l'on sous-estime presque toujours le temps requis pour accomplir une certaine tâche. Nombreux sont les projets qui n'arrivent pas à respecter les échéances prévues. Afin d'éviter les problèmes, une période tampon de dix heures est prévue dans l'échéancier de ce projet. Ce tampon permettra d'absorber les imprévus ou, dans le cas peu probable où il n'aura pas été utilisé par une autre tâche, de paufiner les solutions élaborées pour arriver à produire un travail final de qualité.

TAB. 1 – Échéancier

Étapes	Durée (en heures)	Fin approximative
Tour de lecture	20	17 septembre 2006
Environnement de travail	20	1er octobre 2006
Prototype	30	22 octobre 2006
Améliorations	25	8 novembre 2006
Rédaction du rapport	30	29 novembre 2006
Période tampon	10	5 décembre 2006
Total	135	5 décembre 2006

3.2 Échéancier

Afin de s'assurer de la faisabilité des objectifs de ce projet, il est important de définir un échéancier qui permettra également de s'assurer que la progression du projet suit bien ce qui a été planifié.

Comme il doit s'agir d'un projet de 135 heures, la planification proposée considère une moyenne de travail de dix heures par semaine, à compter de la semaine du 4 septembre 2006. La treizième semaine est celle du 27 novembre, ce qui laisse cinq heures pour la semaine du 4 décembre (la remise est le 5 décembre). La moyenne de dix heures par semaine servira à prévoir les dates approximatives de terminaison des différentes étapes. La table 1 donne les durées estimées des différentes étapes ainsi qu'une estimation de la date où cette dernière devrait être terminée.

La planification du travail ainsi que la méthodologie proposée permettront d'atteindre les résultats visés par ce projet de fin d'études.

4 Résultats

Ce chapitre présentera les différentes étapes du développement de ce projet de fin d'études. Pour chaque méthode ou technique implantée, une explication théorique sera donnée, ainsi les buts et objectifs de l'utilisation de cette méthode. S'il y a lieu, les détails de l'implantation de cette méthode seront donnés pour finalement arriver aux résultats de la technique.

4.1 Premiers pas : simplement Fourier

Avant même de commencer ce travail de recherche, l'auteur avait quelques idées de solutions simplistes au problème de transcription automatique de musique qui devraient fonctionner sur des instances simples du problème. La première partie de ce travail a donc été de tester ces idées.

4.1.1 Explications

En effet, la première méthode proposée était de séparer le son en segments de n échantillons, n étant un paramètre à fixer. Ensuite, on applique la transformée de Fourier à chacun des segments pour identifier, pour chacun des segments, la fréquence la plus forte. À partir de cette information, on est en mesure de déterminer à quelle note correspond cette fréquence et ainsi obtenir éventuellement une liste de notes, avec leur durée.

D'une manière un peu plus détaillée, l'heuristique utilisée pour déterminer le début et la fin d'une note en est une très simple, ayant toutefois un certain nombre d'inconvénients. On fixe un certain seuil d'intensité en deça duquel une fréquence n'est pas considérée comme une note jouée. Ainsi, si la fréquence d'intensité maximale d'un segment a une intensité inférieure à ce seuil, on considère qu'aucune note n'était jouée durant ce segment et qu'on est seulement en présence de bruit. Si la fréquence est bel et bien une note jouée, on compare avec la note jouée au cours du segment précédent. S'il s'agit d'une note différente, on considère qu'une nouvelle note commence à cet instant. S'il s'agit de la même note, on ajoute la durée d'un segment à la durée de cette note déjà créée précédemment. Cette méthode, quoique très simple, possède de nombreux désavantages :

- Si la même note est jouée deux fois de suite sans silence entre les deux attaques, cette méthode ne détectera qu’une seule note d’une longue durée. Dans le cas de test simple que l’on considère, où chaque note est séparée des autres par un silence, ce n’est pas problématique.
- Le choix du seuil d’intensité minimale pour qu’une note soit considérée a une grande influence sur la transcription : un seuil trop élevé masquera certaines notes que la transcription ignorera, alors qu’un seuil trop faible entraînera la détection de notes qui n’en sont pas, qui ne sont que du bruit.
- On ne peut pas détecter la polyphonie. L’heuristique telle que présentée ici ne peut détecter que les mélodies monophoniques. Il serait toutefois facile de le modifier : il suffirait de maintenir une liste de notes « actives » plutôt que de n’en maintenir qu’une seule. Plusieurs fréquences pourraient passer le seuil pour un même segment.

Si on revient à la détection des fréquences en soi, un autre paramètre doit être choisi, soit n , le nombre d’échantillons par segment. Le calcul suivant permet de donner un ordre de grandeur.

La FFT donne comme résultat un vecteur contenant l’intensité des fréquences réparties linéairement de 0 à la moitié du taux d’échantillonnage de la pièce. Le taux choisi pour les tests de ce projet est 44100 Hz, qui est le standard généralement utilisé pour les disques compacts. Comme les fréquences des notes augmentent exponentiellement par rapport aux notes jouées¹⁰, la différence de fréquence entre deux notes séparées d’un demi-ton sera plus faible dans les basses fréquences. La note la plus basse jouable sur une guitare correctement accordée est située 29 demi-tons sous le La 440 Hz. On trouve donc la différence minimale entre deux notes que l’on veut être en mesure de distinguer :

$$\Delta_{min} = (440 \times 2^{\frac{-29}{12}}) - (440 \times 2^{\frac{-28}{12}}) = 4.90016863\dots$$

Si n est le nombre d’échantillons d’un segment, la différence entre les fréquences de la représentation après la transformée de Fourier est donnée par $\frac{22050}{n}$. On met un facteur de deux à la différence minimale trouvée plus

¹⁰La fréquence double à chaque octave. Par exemple, le premier La jouable sur une guitare est à 110 Hz. Le suivant est à 220 Hz, puis l’autre à 440 Hz, 880 Hz, etc.

haut afin de pouvoir correctement associer une fréquence à une note. On obtient ainsi la formule suivante :

$$\frac{\frac{22050}{2}}{\frac{n}{2}} = \frac{4.90016863...}{2} \Rightarrow n = 2 \left(22050 \frac{2}{4.90016863...} \right) = 17999.3806...$$

On devrait donc avoir besoin de segments d'au moins 18 000 échantillons. À noter qu'à une fréquence d'échantillonnage de 44100 Hz, ce nombre d'échantillons correspond à plus de quatre dixièmes de seconde. Un des désavantages de cette méthode est qu'étant donné que la FFT donne les fréquences sur une échelle linéaire alors que l'échelle des notes est logarithmique, on obtient une plus grande précision dans les hautes fréquences que dans les basses.

4.1.2 Objectifs

Malgré les différents problèmes mentionnés plus tôt, cette méthode devrait permettre d'obtenir une première transcription automatique pour un cas très simple. Il est à noter que l'on s'attend à obtenir de meilleurs résultats pour une mélodie jouée dans les hautes fréquences que pour une jouée dans les basses fréquences. Étant donné qu'elle utilise la FFT et qu'elle ne fait qu'une simple recherche de maximum, elle devrait être assez performante.

4.1.3 Détails d'implantation

La majorité des détails d'implantation concernaient les algorithmes et heuristiques utilisés et ont été décrits plus haut. À noter toutefois qu'il était très pratique de pouvoir facilement ajuster les paramètres comme le seuil d'acceptation d'une fréquence ainsi que la taille des segments afin de pouvoir obtenir des résultats satisfaisants.

Aussi, afin de permettre de déboguer plus facilement certains types de problème, un affichage graphique fort utile a été implanté. L'axe des X représente le temps et l'axe des Y les fréquences, puis pour chacun des segments temporels, on représente l'intensité de chacune des fréquences à l'aide d'une couleur : bleu pour 0 et rouge pour l'intensité maximale. Certains de ces graphiques seront reproduits plus bas pour mieux expliquer certains résultats.

TAB. 2 – Notes du son généré comme entrée du premier système

Note	Début	Durée
72	1.0	0.4
72	1.5	0.4
72	2.0	0.4
74	2.5	0.4
76	3.0	0.9
74	4.0	0.9
72	5.0	0.4
76	5.5	0.4
74	6.0	0.4
74	6.5	0.4
72	7.0	0.9

TAB. 3 – Notes transcrites par le premier système

Note	Début	Durée
72	1.02168	0.371519
72	1.48608	0.371519
72	2.04336	0.371519
74	2.50776	0.371519
76	2.97215	0.928798
74	3.99383	0.928798
72	5.01551	0.371519
76	5.47991	0.464399
74	6.03719	0.371519
74	6.50159	0.371519
72	6.96599	0.928798

4.1.4 Résultats

L'utilisation de cette technique a permis d'atteindre les résultats espérés, c'est-à-dire qu'on a été en mesure de transcrire une mélodie simple, soit Au Clair de la Lune. La table 2 représente la liste de notes¹¹ qui a servi à générer le son pris comme entrée du système¹² alors que la table 3 donne les résultats produits par le système. On a obtenu avec cet exemple une identification parfaite des notes, avec une identification du début et de la durée des notes précise à quelques centièmes de secondes près. Le seuil de passage utilisé était la moitié de l'intensité maximale rencontrée.

4096 échantillons par segment ont été utilisés pour effectuer cette transcription. On voit sur la figure 2 la répartition de l'intensité des fréquences en fonction du temps et de la fréquence (de 0 à 690 Hz sur l'axe vertical, ce qui est suffisant pour cette pièce). Si on compare avec la transcription de la même pièce avec 8192 échantillons par segment, on obtient la figure 3. L'augmentation du nombre d'échantillons par segment entraîne une augmentation de la résolution fréquentielle, mais une diminution de la résolution temporelle, ce qui est logique car on subdivise plus finement l'intervalle de fréquences allant de 0 à la moitié du taux d'échantillonnage, au prix d'une augmentation du temps que représente chacun des segments. Cette seconde transcription, non représentée ici, est beaucoup moins précise au niveau du temps, regroupant même certaines notes de la même hauteur malgré le silence qui les sépare.

Dans le même ordre d'idée, si on utilise un nombre d'échantillons par segment deux fois moins élevé, on obtient le résultat inverse. En effet, la transcription est plus précise quant au début et la fin des notes, mais elle se trompe quant à la hauteur de certaines notes. Il s'agit là d'une conclusion importante applicable à de nombreux autres problèmes de traitement de signaux numériques : en général, lorsque le nombre d'échantillons par segment augmente, la résolution temporelle diminue alors que la résolution fréquentielle augmente.

¹¹Chaque note est représentée par un entier. 69 correspond au La 440 Hz et l'ajout ou le retrait de 1 au numéro de la note correspond à une augmentation ou une diminution de un demi-ton.

¹²La pièce a été générée par CSound à l'aide d'un instrument jouant chaque note comme un signal sinusoïdal pur.

FIG. 2 – Première transcription à 4096 échantillons par segment

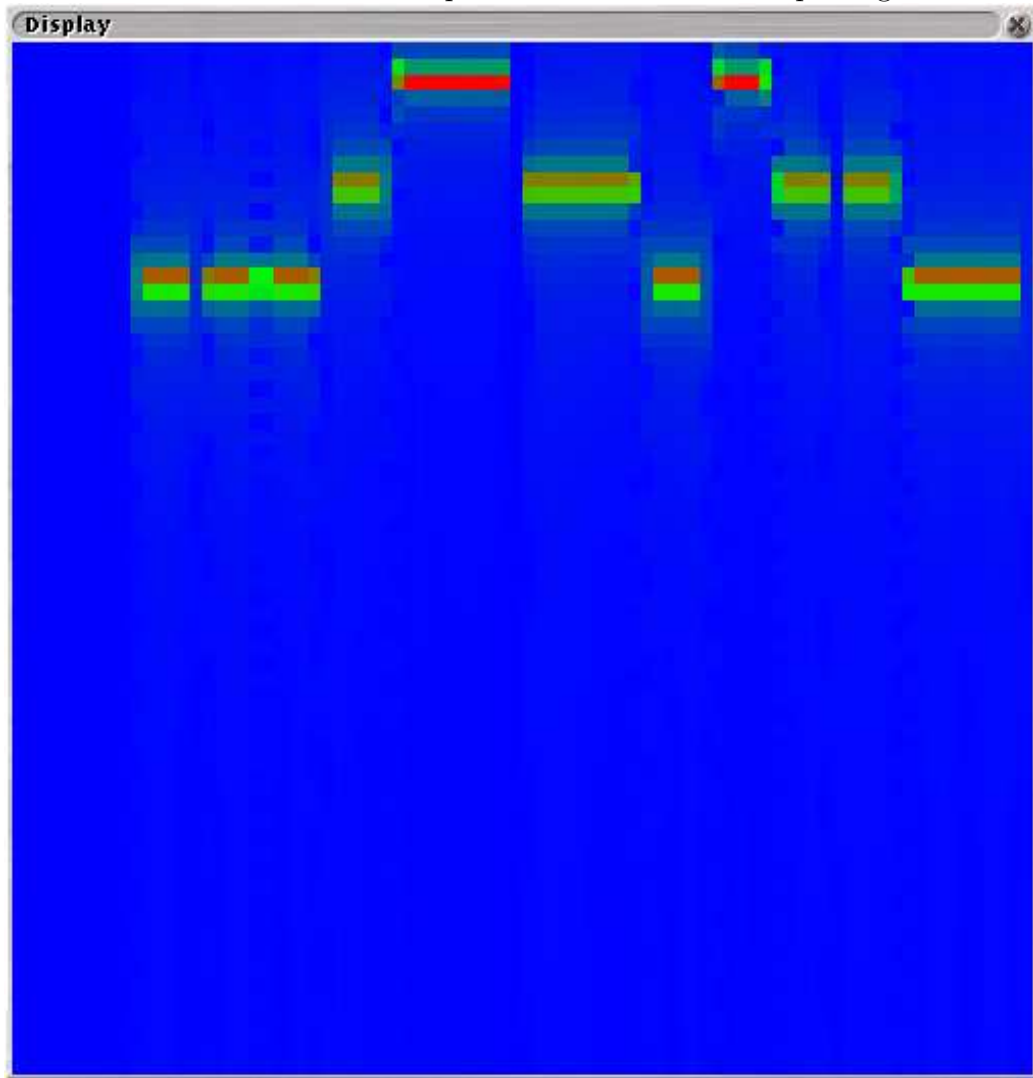
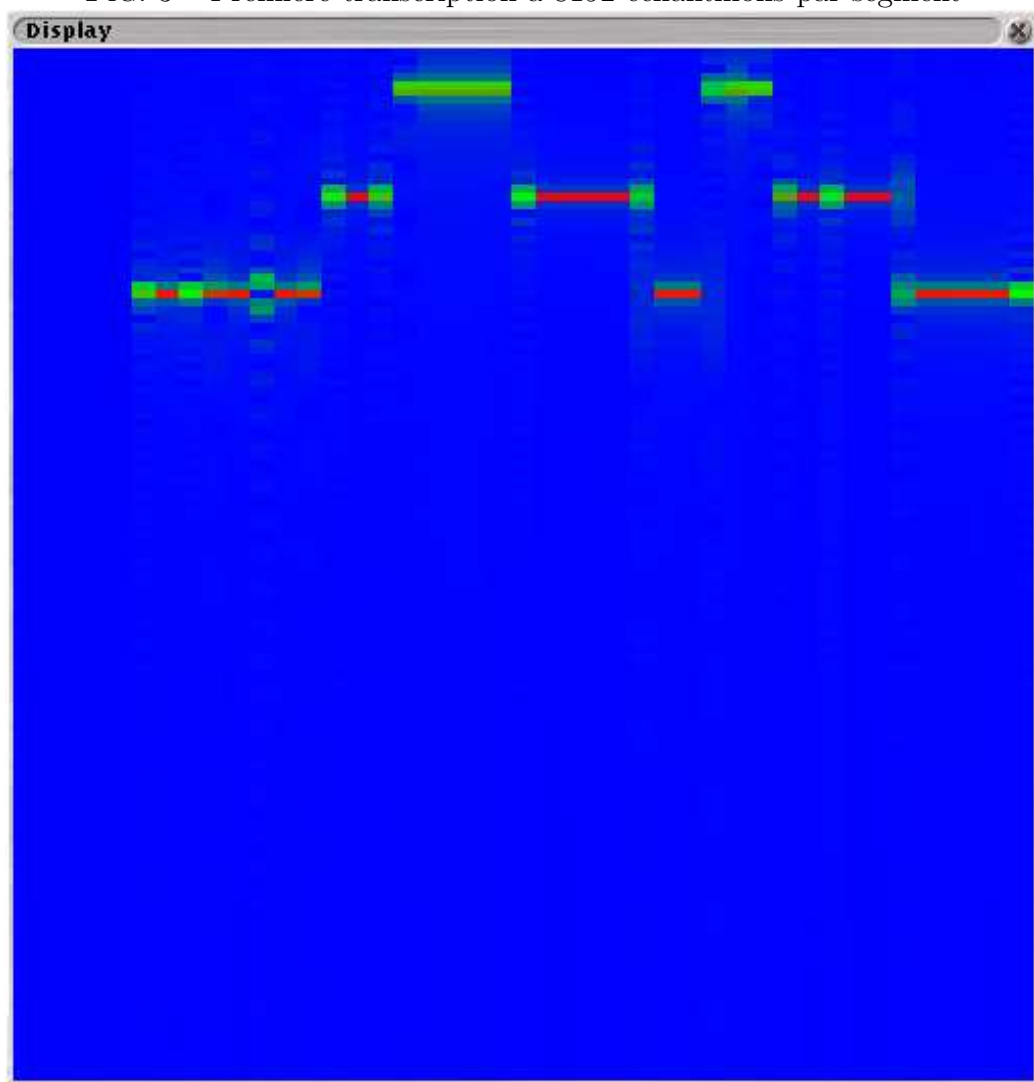


FIG. 3 – Première transcription à 8192 échantillons par segment



Tel que mentionné précédemment, un des problèmes attendus de cette méthode est sa moins grande précision pour les basses fréquences. En effet, si on tente la transcription de la même mélodie, mais un octave plus bas (on réduit donc la fréquence de chaque note de moitié), on obtient des problèmes d'identification de notes, ce qui est relié à la résolution temporelle. La prochaine technique visera donc à améliorer cette lacune.

4.2 Échelle logarithmique

Avant de tenter de passer à la transcription de pièces enregistrées plutôt que générées, l'auteur a tenté de régler un des problèmes du prototype précédent, c'est-à-dire celui de la précision moindre pour les fréquences moins élevées.

4.2.1 Explications

L'idée de base est de simplement mieux choisir la répartition des fréquences de la représentation dans le domaine des fréquences : plutôt que de choisir une distribution linéaire des fréquences comme le fait la FFT, il s'agit de choisir une distribution logarithmique afin d'avoir moins d'espace entre les fréquences dans la partie basse du spectre. Ainsi, comme l'échelle des fréquences associées aux notes est logarithmique, on retrouvera la même densité de fréquences dans un intervalle musical¹³ donné, peu importe sa position dans la gamme des fréquences.

En théorie, comme avec cette technique on choisit vraiment les fréquences que l'on désire, on pourrait ne trouver que l'intensité des fréquences associées à chacune des notes de la guitare, soit entre 40 et 50. Toutefois, pour plus de précision et pour contrer les différentes sources de distortion, on subdivise l'intervalle entre deux notes adjacentes (séparées par un demi-ton) selon un certain facteur, par exemple 16.

Le reste de la procédure est extrêmement semblable à la technique précédente : la même heuristique d'identification du début et de la fin d'une note est utilisée, et on arrive avec le même type de liste de notes comme sortie.

¹³On entend par intervalle musical une différence entre deux notes, comme une seconde ou une quinte, qui peut également se compter en demi-tons.

TAB. 4 – Notes du son généré comme entrée du premier système modifié

Note	Début	Durée
60	1.0	0.4
60	1.5	0.4
60	2.0	0.4
62	2.5	0.4
64	3.0	0.9
62	4.0	0.9
60	5.0	0.4
64	5.5	0.4
62	6.0	0.4
62	6.5	0.4
60	7.0	0.9

4.2.2 Objectifs

Tel que mentionné plus haut, cette technique vise à modifier la répartition des fréquences dans la représentation d'une partie du signal dans le domaine des fréquences afin de conserver la même précision, peu importe la hauteur dans la gamme des fréquences. En effet, comme la différence de fréquence, en Hz, entre deux notes séparées d'un demi-ton est beaucoup plus faible dans les basses fréquences que dans les hautes, une distribution linéaire entraîne un déséquilibre dans la précision, que cette technique tente de corriger.

4.2.3 Détails d'implantation

Comme on spécifie chacune des fréquences dont on veut connaître l'intensité dans un segment de signal, il n'est malheureusement pas possible d'utiliser la FFT pour cette technique. Une implantation très simple d'une DFT a donc été réalisée.

4.2.4 Résultats

Les résultats de cette technique correspondent tout-à-fait à ce qu'on attendait. Elle permet en effet de transcrire la pièce, même si elle est jouée à

TAB. 5 – Notes transcrites par le premier système modifié

Note	Début	Durée
60	0.989751	0.415057
60	1.50059	0.383129
60	2.01143	0.383129
62	2.49034	0.415057
64	3.00118	0.893968
62	3.99093	0.893968
60	5.01261	0.383129
64	5.49152	0.415057
62	6.00236	0.383129
62	6.5132	0.383129
60	6.99211	0.893968

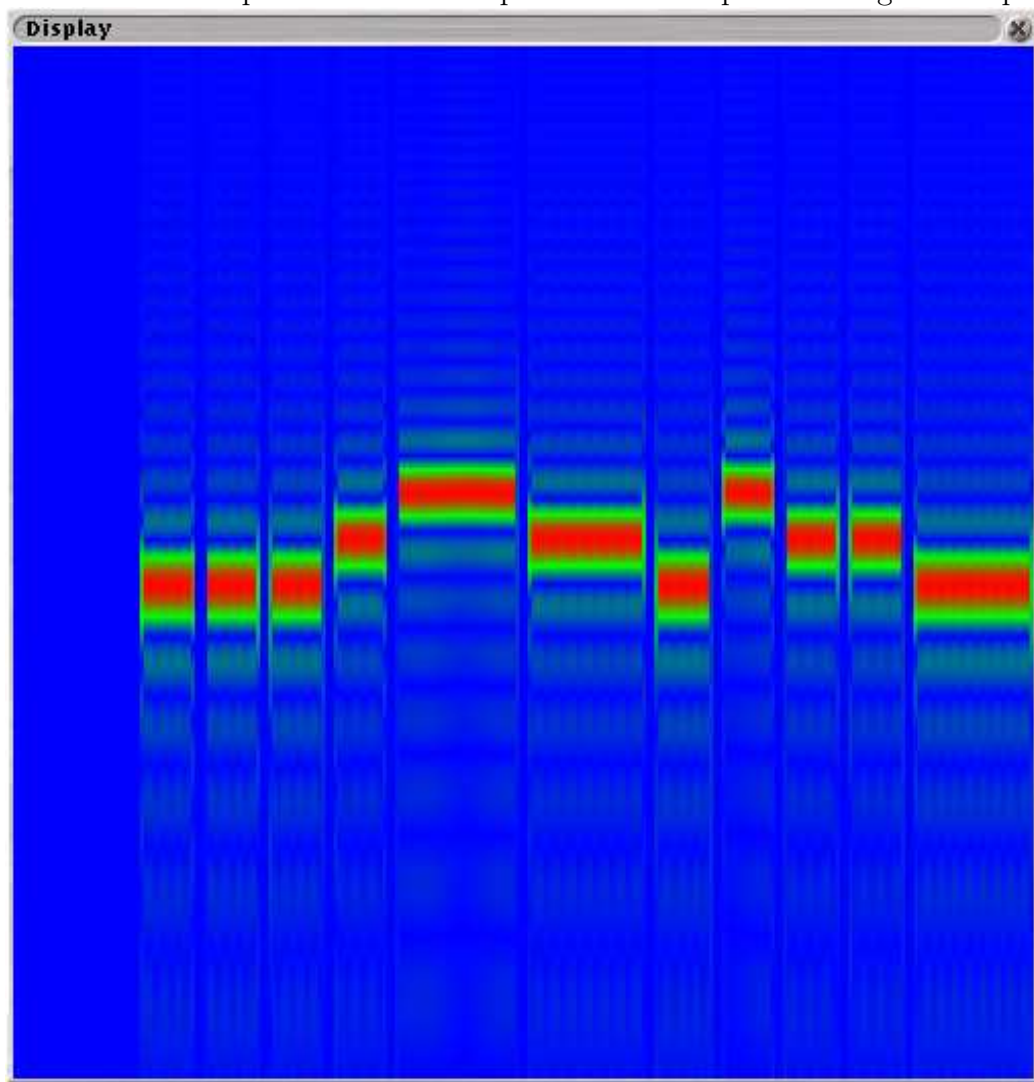
un octave plus bas¹⁴ que pour la méthode précédente, comme on peut le voir sur les tables 4 et 5.

La figure 4 montre que l’identification de la hauteur des notes, ainsi que le début et la fin, semble beaucoup plus précis. Cette figure représente toutes les fréquences testées (représentées sur une échelle logarithmique verticale), plutôt que seulement une petite partie comme c’était le cas lorsque la distribution des fréquences était linéaire. Étant donné qu’on ne calcule pas d’intensité de fréquences qui ne sont pas pertinentes, comme dans le cas d’une distribution linéaire, moins d’échantillons par segment sont nécessaires pour arriver à une résolution fréquentielle intéressante, ce qui explique l’augmentation de la résolution temporelle. De plus, comme la distribution des fréquences calculées suit à présent la distribution des fréquences des notes, on arrive également à une meilleure résolution fréquentielle, comme le montre la plus grande netteté de la figure 4 par rapport aux figures 2 et 3.

Toutefois, comme on pouvait s’y attendre, la performance de cette méthode ne se compare pas à celle utilisant la FFT : on parle d’un facteur d’environ 100 pour la transcription prise en exemple, soit *Au Clair de la Lune*. Il pour-

¹⁴En fait, avec cette technique, *Au Clair de la Lune* a pu être transcrit lorsqu’il était joué dans la tonalité la plus basse qu’il est possible d’atteindre sur une guitare accordée normalement.

FIG. 4 – Transcription à l'aide de représentation fréquentielle logarithmique



rait être intéressant de combiner cette technique avec une autre, plus rapide, afin de peut-être préciser les résultats, sans pour autant avoir à subir la baisse de performance énorme qu'entraîne l'utilisation de cette technique.

Cette méthode permet toutefois d'atteindre le premier objectif, soit l'obtention d'un prototype capable de transcrire les cas les plus simples, ayant même subi une amélioration lui permettant de traiter les cas plus complexes où la pièce est jouée dans de basses fréquences. Le prochain facteur de difficulté qui sera ajouté sera la transcription de guitare enregistrée, qui entraînera certes de nouveaux problèmes.

4.3 Guitare enregistrée

Le but final de ce projet est d'arriver à transcrire des pièces musicales jouées à la guitare. Il est donc maintenant temps de s'attaquer à un enregistrement de guitare.

4.3.1 Explications

Cette étape constitue en fait une phase d'expérimentation : aucune nouvelle méthode ne sera à priori implantée. À partir du prototype déjà implanté, on se contentera de procéder aux ajustements nécessaires afin d'arriver à transcrire correctement un son enregistré.

4.3.2 Objectifs

Le but de cette étape est tout simplement de transposer à des pièces jouées à la guitare les résultats déjà obtenus sur des pièces générées.

4.3.3 Détails d'implantation

Toujours pour les mêmes raisons de simplicité du cas, la première pièce à avoir été enregistrée est *Au Clair de la Lune* joué à la même hauteur de fréquence que la première pièce générée qui a été transcrite (voir table 2). Les premiers enregistrements ont été faits en branchant directement une guitare électrique dans la carte de son d'un ordinateur, l'amplification étant faite en logiciel. Le son enregistré contenait un taux élevé de bruit, ce qui pourra avoir une influence sur les résultats obtenus.

Seuls les premiers tests ont été réalisés avec ces enregistrements de mauvaise qualité. Par la suite, un système d'enregistrement de niveau professionnel, soit le FirePod de PreSonus, a été utilisé, réduisant considérablement le bruit présent dans les enregistrements.

4.3.4 Résultats

La figure 5 montre le résultat de la transcription à l'aide de la première méthode, c'est-à-dire avec une distribution linéaire des fréquences, sur un signal ayant été normalisé (voir l'annexe D pour une explication des effets de l'absence de cette normalisation). Malgré l'importante présence de bruit, la transcription a donné des résultats pratiquement équivalents à ceux de la transcription d'un son généré. Un résultat comparable est donné par l'utilisation de la méthode logarithmique. On remarque toutefois un phénomène qui n'était pas visible lors de la transcription des pièces générés : la force d'une note diminue avec le temps lorsqu'elle est maintenue. En effet, on voit clairement, sur la figure 6, que la couleur rouge d'une note, synonyme d'une forte intensité, s'estompe peu à peu.

Une autre pièce a été soumise au système à cette étape, soit une partie de Rondo Alla Turca de Mozart (voir Annexe E) jouée à environ 5,33 notes par secondes¹⁵.

Les résultats obtenus n'ont pas été aussi « parfaits » qu'avec Au Clair de la Lune, même avec la méthode logarithmique. En bref, certaines notes n'ont pas été identifiées, alors que des notes superflues, souvent de quelques centièmes de secondes, ont été trouvées. Ces brèves notes se trouvaient souvent à la frontière de notes, leur détection ayant probablement été causée par l'analyse des fréquences d'un segment contenant le début de la note, ne possédant donc pas une fréquence fondamentale propre.

Comme la transcription fonctionne désormais pour des cas simples d'enregistrement de guitare, l'objectif sera désormais d'améliorer la justesse de la transcription de pièces plus difficiles.

¹⁵Cette pièce a été jouée avec comme référence un métronome battant à 80 coups par minute. Comme la partie jouée est exclusivement composée de double-croches, l'extrait comportait 320 notes par minute ou environ 5,33 notes par seconde.

FIG. 5 – Transcription d'une pièce avec beaucoup de bruit

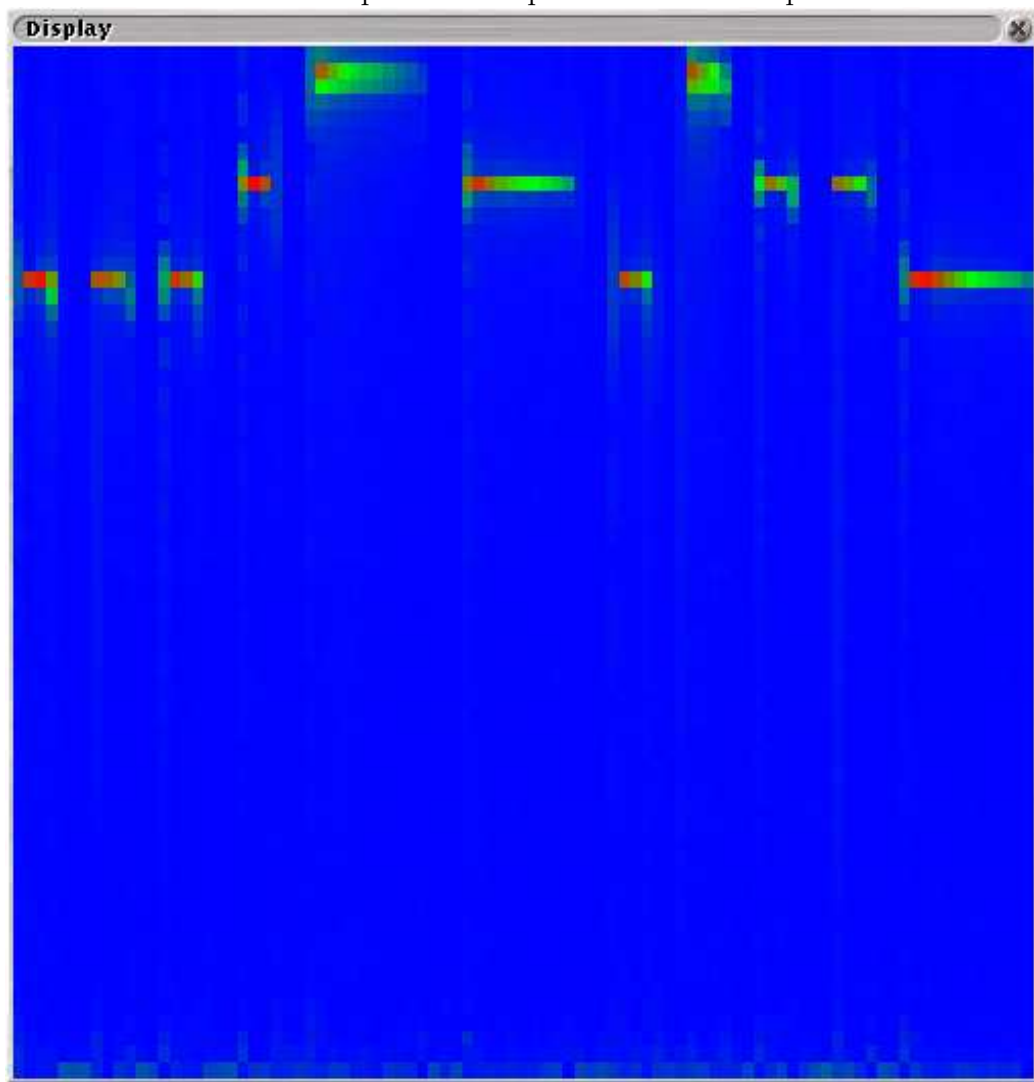
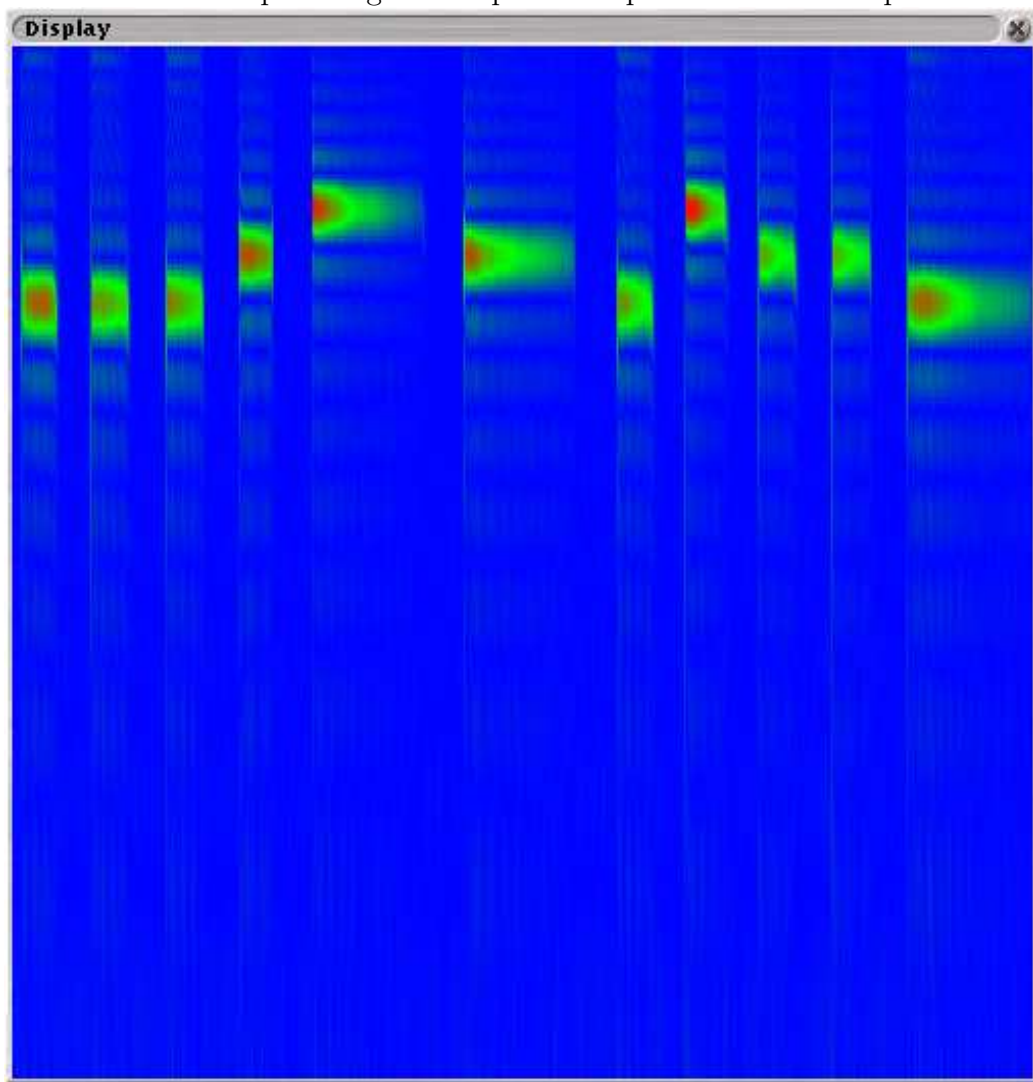


FIG. 6 – Transcription logarithmique d'une pièce avec beaucoup de bruit



4.4 Ajout de zéros

La prochaine technique en est une qui est couramment utilisée, conjointement avec la FFT, pour identifier avec plus de précision les pics des fréquences reconnues.

4.4.1 Explications

La FFT effectuée sur n échantillons donne comme résultats les $\frac{n}{2} + 1$ fréquences réparties également entre 0 et la moitié de la fréquence d'échantillonnage du signal analysé. Si on désire augmenter la résolution fréquentielle de la transformée de Fourier, un des moyens consiste à augmenter le nombre d'échantillons dans le segment. On a toutefois vu que cette méthode entraîne une diminution équivalente de la résolution temporelle, ce qui n'est évidemment pas souhaitable.

La technique proposée ici consiste plutôt à ajouter au segment des échantillons de valeur nulle afin d'augmenter le nombre d'échantillons passés à la transformée de Fourier et ainsi d'augmenter le nombre de subdivisions de l'échelle de fréquence entre 0 et la moitié du taux d'échantillonnage. Il est important de noter que cette technique *n'augmente pas* la résolution de la transformée, car on ne peut pas ajouter de la nouvelle information sur les fréquences à partir de ce que l'on a au départ. Il s'agit tout simplement d'une forme d'interpolation entre les valeurs connues de la transformée.

4.4.2 Objectifs

L'utilisation de cette technique possède un avantage majeur : elle permet d'utiliser la FFT et donc d'obtenir une bien meilleure performance qu'avec la DFT sur une échelle logarithmique de fréquences. Aussi, la possibilité d'interpoler entre les valeurs devraient permettre de diminuer le nombre d'échantillons par segment et ainsi obtenir une meilleure résolution temporelle, tout en maintenant une résolution fréquentielle suffisante pour la transcription. Comme cette technique utilise quand même une répartition linéaire des fréquences sur l'échelle des fréquences, on s'attend à de meilleurs résultats dans la partie élevée du spectre des fréquences.

4.4.3 Détails d'implantation

Comme `fftw([9])` est l'engin utilisé pour le calcul de la FFT, l'ajout de zéros à la fin du segment n'a pu être fait sans passer par une copie dans un espace mémoire plus grand, dont toutes les valeurs sont mises à zéro à l'exception des valeurs du segment qui y sont copiées. Le nombre de zéros à ajouter est configurable à l'aide d'un facteur entier correspondant à la nouvelle taille du segment avec des zéros par rapport à celle du segment original. Par exemple, un segment de 128 échantillons deviendra, avec un facteur de 4, un segment de 512 échantillons : 384 zéros ont donc été ajoutés.

4.4.4 Résultats

Les figures 7, 8, 9, 10, 11 et 12 montrent l'effet de l'ajout de zéros sur la détermination de l'importance de chacune des fréquences dans les différents segments. On remarque que plus l'interpolation est fine, mieux semble défini le contour de chaque note. Grâce à cette subdivision plus fine, on est maintenant en mesure de transcrire correctement les versions générées de *Au Clair de la Lune* que l'on arrivait pas à transcrire étant donné qu'elles étaient jouées trop basses. De plus, la transcription d'une partie de *Rondo Alla Turca* de Mozart, jouée à environ 5,33 notes par secondes, surpasse de loin les transcriptions précédentes. Seulement quelques notes n'ont été transcrites (notes absentes), et on trouve parfois, au début ou à la fin d'une note, une note incorrecte d'une durée égale à la longueur d'un segment de base pour l'analyse de la FFT, soit la plus petite unité de longueur possible avec l'analyse faite (environ 0,02 seconde). Cette erreur s'explique par la tentative d'assigner une note à un segment qui tombe en plus sur l'attaque d'une note, où le signal varie énormément. L'ajout d'un critère de sélection supplémentaire permettrait d'éliminer facilement ce type de problème. Ainsi, les résultats de la technique d'ajout de zéros sont très encourageants, car ils donnent une bonne qualité de transcription et ce avec une bonne performance.

En ignorant ces notes trop courtes (qui, de toute façon, entraînent des problèmes de génération de son avec `CSound`) et en ajustant les résultats, on obtient une transcription presque parfaite, c'est-à-dire qui n'oublie que deux notes. Si on augmente la difficulté, c'est-à-dire qu'on tente de transcrire la même mélodie jouée à 100 coups par minute (6.666 notes par seconde) ou à 120 coups par minute (8 notes par seconde), les erreurs s'accroissent :

davantage de notes sont oubliées et d'autres sont identifiées incorrectement.

Par ailleurs, un problème a été mis en évidence par la tentative de transcription de certains échantillons enregistrés avec le moins de bruit possible. On remarque sur la figure 13 que pour chaque note jouée (dans le bas de la figure), on a plusieurs pics de fréquences plus haut sur le spectre de fréquences. Ce problème, que l'on ne retrouve pas avec des pièces générées avec un son sinusoïdal pur, apparaît toutefois lorsque l'on s'attaque à la transcription d'instruments réels. Dans ce cas particulier, la transcription de *Au Clair de la Lune* est faussée par la détection de notes de fréquences beaucoup plus élevées à cause de la présence de ces harmoniques d'ordres supérieurs.

Il pourrait être intéressant de détecter ces cas où une note n'est en fait qu'un mode de vibration supérieur d'une note fondamentale plus basse. Cela demande une certaine connaissance de l'importance relative des harmoniques de l'instrument analysé, car la forme spectrale des harmoniques est un des éléments qui distinguent les instruments les uns des autres. Toutefois, la fixation d'un seuil d'acceptation reviendrait à choisir un compromis entre la détection de notes « fantômes » à l'octave, qui ne sont pas vraiment là, et le rejet de notes à l'octave qui ont été jouées mais qui se retrouvent sous le seuil.

4.5 Utilisation de fenêtres non rectangulaires

Depuis le début de l'expérimentation avec les différentes techniques utilisant la transformée de Fourier, chacun des segments analysés a été soumis à la FFT sans être préalablement transformé. Il est toutefois possible d'appliquer plutôt une *fenêtre* sur les échantillons qui seront analysés et ce concept est à la base de la technique qui sera ici décrite.

4.5.1 Explications

Lorsque l'on analyse un segment d'un signal correspondant à une pièce musicale en le soumettant à la transformée de Fourier, cette dernière le traite comme si le segment représentait une période d'un signal périodique. Or, si une des fréquences du signal analysé n'est pas un multiple de la fréquence de base correspondant à la longueur du segment, certains artéfacts se pro-

duisent, comme on peut le remarquer dans l'exemple suivant.

La figure 14 montre un signal de 8 Hz échantillonné à 100 Hz. La partie de gauche montre la représentation dans le domaine du temps du signal : on voit qu'il commence et termine « à la même phase », c'est-à-dire au même endroit à l'intérieur de sa période, pouvant donc être bouclé sans problème. Cela se traduit par une seule valeur non nulle dans la partie droite de la figure qui illustre le domaine des fréquences¹⁶. Par contre, la figure 15 montre un signal très semblable, mais dont la fréquence est 8.5 Hz. Contrairement au signal de la figure 14, il ne se termine pas au même endroit à l'intérieur de sa période, ce qui signifie que si on bouclait ce signal, il y aurait une discontinuité de la phase à chaque boucle. On peut voir dans la partie droite de la figure que cela a pour conséquence un écoulement spectral¹⁷, c'est-à-dire que le pic correspondant à la fréquence « s'écoule » sur les fréquences avoisinantes.

Afin de contrer ce problème, on peut multiplier le segment analysé par un autre signal que l'on appelle *fenêtre*, c'est-à-dire un signal qui est nul avant un certain temps et après un certain temps, avec une certaine forme non nulle entre ces deux temps. La multiplication du segment par une fenêtre permet, selon la forme de la fenêtre, de minimiser l'impact des échantillons près du début et de la fin du segment. À noter que la multiplication de deux signaux dans le domaine du temps correspond à une convolution dans le domaine des fréquences. Une des fenêtres les plus couramment utilisées est la fenêtre de Hamming et est donnée par l'équation suivante, où $H(i)$ est la valeur du signal de fenêtre à l'échantillon i et N est la taille du signal de fenêtre, qui est égale à la taille du segment à fenêtrer :

$$H(i) = 0.54 - 0.46 \cos\left(\frac{2\pi i}{N-1}\right)$$

La figure 16 montre, dans le domaine du temps ainsi que dans celui des fréquences, le signal de la figure 15 après avoir été fenêtré par une fenêtre de Hamming. On remarque donc une certaine réduction de l'effet d'écoulement spectral. Il est à noter que la simple copie d'une partie d'un signal pour obtenir le segment correspond à une fenêtre rectangulaire, c'est-à-dire nulle avant un certain temps, nulle après un certain temps et possédant une valeur de 1

¹⁶La forme de triangle n'est qu'une illusion : le seul point non nul est lié par une ligne aux points nuls situés de chacun de ses côtés, ce qui donne la forme de pic en triangle.

¹⁷Traduction libre de *spectral leakage*.

entre ces deux temps.

Ainsi, la fenêtre de Hamming sera appliquée aux segments passés à la transformée de Fourier et ce en combinaison avec la méthode de l'ajout de zéros, qui s'est avérée la meilleure jusqu'à présent.

4.5.2 Objectifs

On vise, par l'application de cette méthode, à réduire les artéfacts comme l'écoulement spectral afin d'obtenir une meilleure transcription. Cette technique permettra peut-être d'éliminer certaines notes « bruit » qui étaient faussement identifiées, ce qui permettra peut-être par la suite d'utiliser des paramètres de transcription plus « agressifs », mais qui n'ont pas été utilisés jusqu'à présent à cause de leur plus grande susceptibilité de rapporter des erreurs.

4.5.3 Détails d'implantation

Pour implanter cette technique, il a suffi de remplacer la copie des échantillons du signal original à l'espace mémoire allouée pour l'ajout de zéros par une opération de fenêtrage. Alternier entre une fenêtre rectangulaire et une fenêtre de Hamming, les seules qui ont été testées dans le cadre de ce projet, est trivial.

4.5.4 Résultats

Les résultats de cette méthode sont supérieurs à ceux de la transcription sans utilisation de fenêtre. En effet, pour une partie de Rondo Alla Turca jouée à 100 coups par minute, seulement quelques notes n'ont pas été retranscrites, alors que pour la version jouée à 120 coups par minute, à ces quelques notes omises s'ajoute une note de trop à un octave supérieur.

Par contre, dans les deux cas, un nouveau type d'erreur s'est présenté. Effectivement, certaines notes ont été transcrites comme ayant été coupées en deux. Par exemple, une note d'une durée de 120 millisecondes a été séparée en deux notes de la même hauteur, d'une durée de 50 millisecondes chacune et séparées d'un silence de 20 millisecondes. Musicalement, il s'agit d'un remplacement d'une double-croche de la mélodie par deux triples-croches de la

même note.

Cette séparation est causée par une erreur d'évaluation d'un seul segment, car la durée du silence entre les notes est égale à la durée d'un segment. La note de ce segment est rejetée, car son intensité tombe sous le seuil qui différencie une note de bruit. L'abaissement de ce seuil corrige le problème, mais entraîne la détection d'autres notes qui n'en sont pas réellement.

Ainsi, avec la combinaison de l'ajout de zéros et de l'utilisation d'une fenêtre, on arrive à des résultats intéressants pour la transcription d'une partie de Rondo Alla Turca jouée à 80, 100 ou 120 coups par minute.

4.6 Autres...

Ici s'arrête l'implantation de techniques de traitement de signaux numériques pouvant aider à la transcription automatique de musique. Certaines autres méthodes auraient pu être implantées assez rapidement si le temps l'avait permis.

D'abord, il aurait été facile d'améliorer l'algorithme de détection et de classification des notes. Bien qu'il ait été amplement suffisant pour la réalisation des tests effectués dans le cadre de ce travail, il aurait été possible de le faire traiter plus adéquatement certains cas, comme la répétition d'une même note sans silence pour séparer les différents coups ou la détermination de si un segment est une note ou du bruit.

Par ailleurs, il aurait également été intéressant d'explorer une méthode n'utilisant pas la transformée de Fourier, mais travaillant plutôt uniquement dans le domaine du temps (voir [18]). Cette méthode vise à reconnaître, dans la variation du signal en fonction du temps, une certaine périodicité du signal pour réussir à en identifier la période et donc la fréquence. Le temps a toutefois manqué pour tenter d'implanter cette technique.

En somme, même sans avoir pu implanter les techniques précédentes, une multitude de méthodes ont été explorées, pour arriver finalement à être capable de produire automatiquement une transcription que l'on peut qualifier « d'acceptable » d'une pièce interprétée à la guitare dont le rythme est assez rapide.

FIG. 7 – Ajout de zéros (facteur 1)

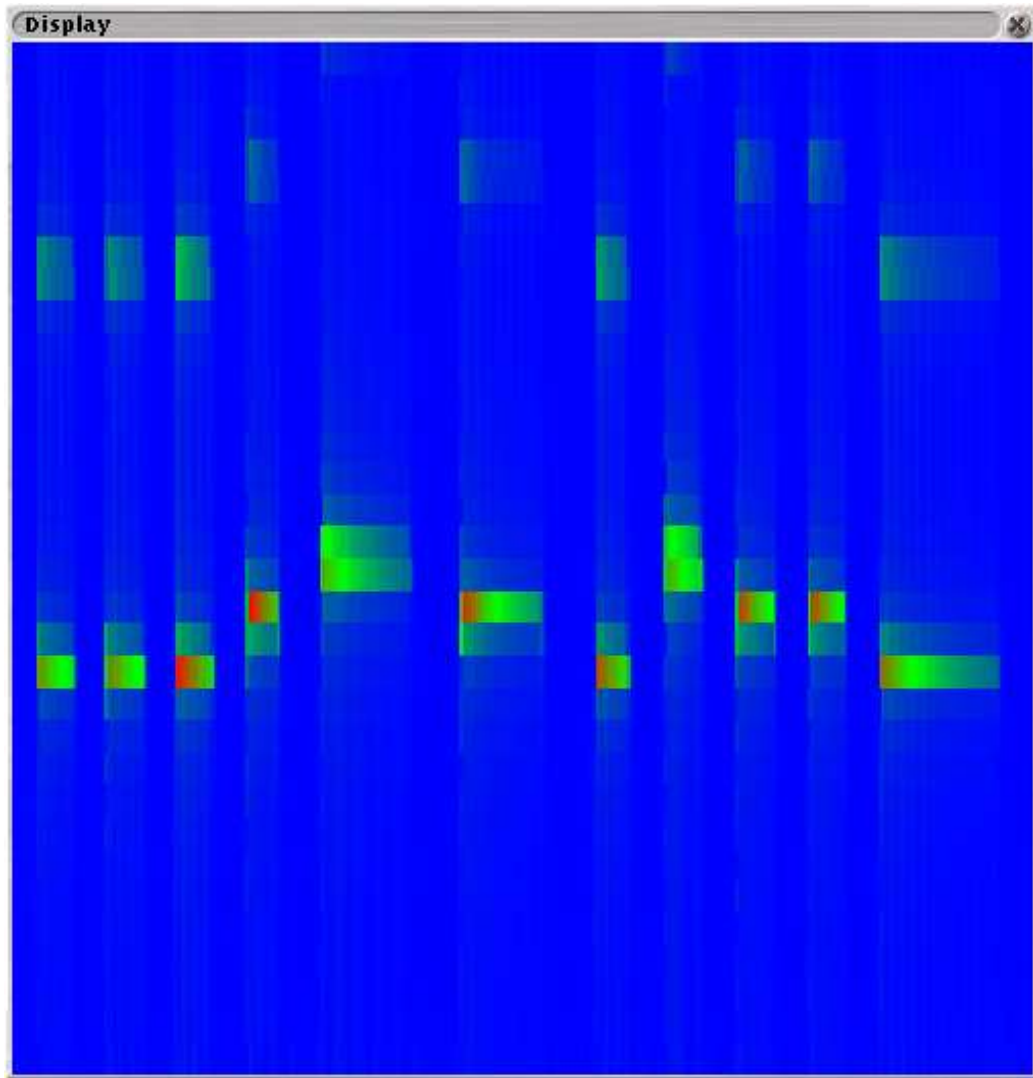


FIG. 8 – Ajout de zéros (facteur 2)

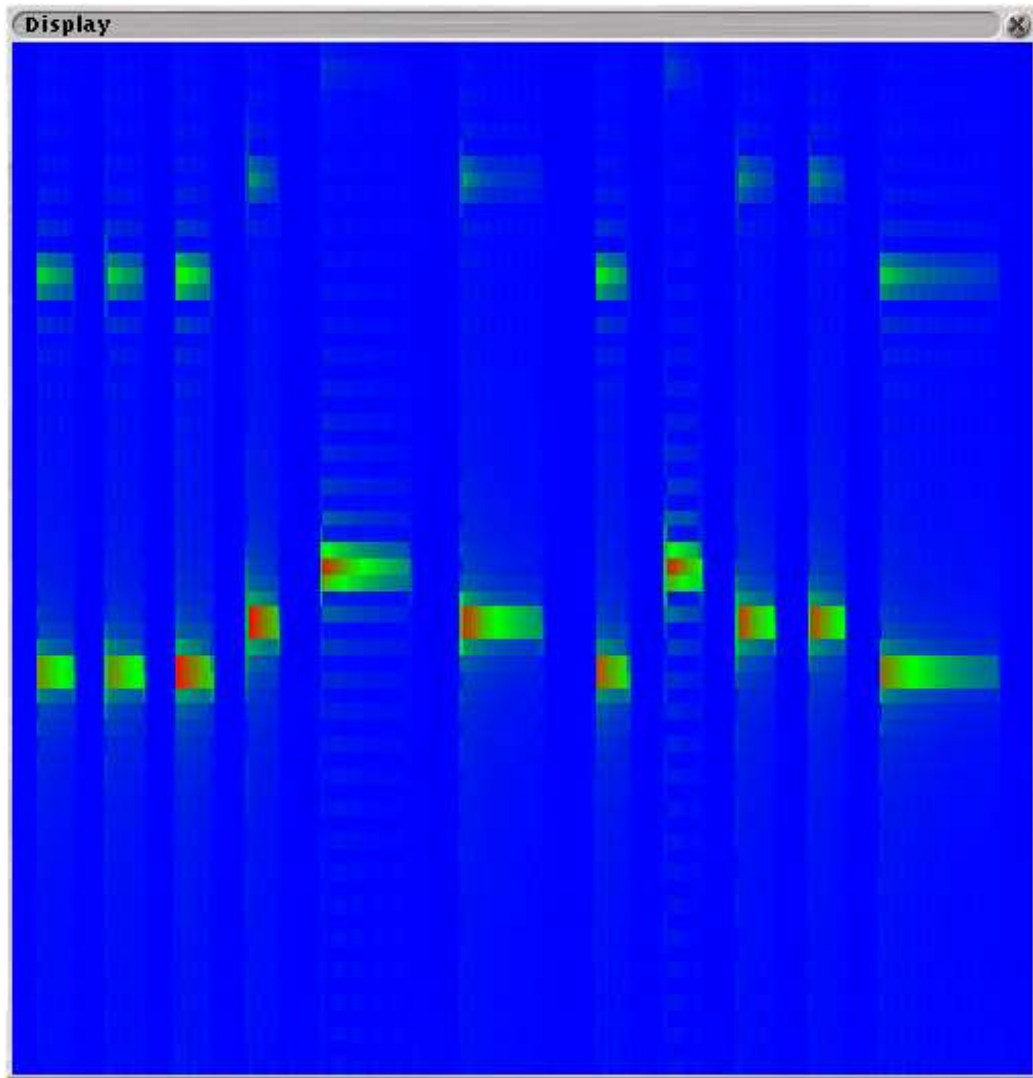


FIG. 9 – Ajout de zéros (facteur 4)

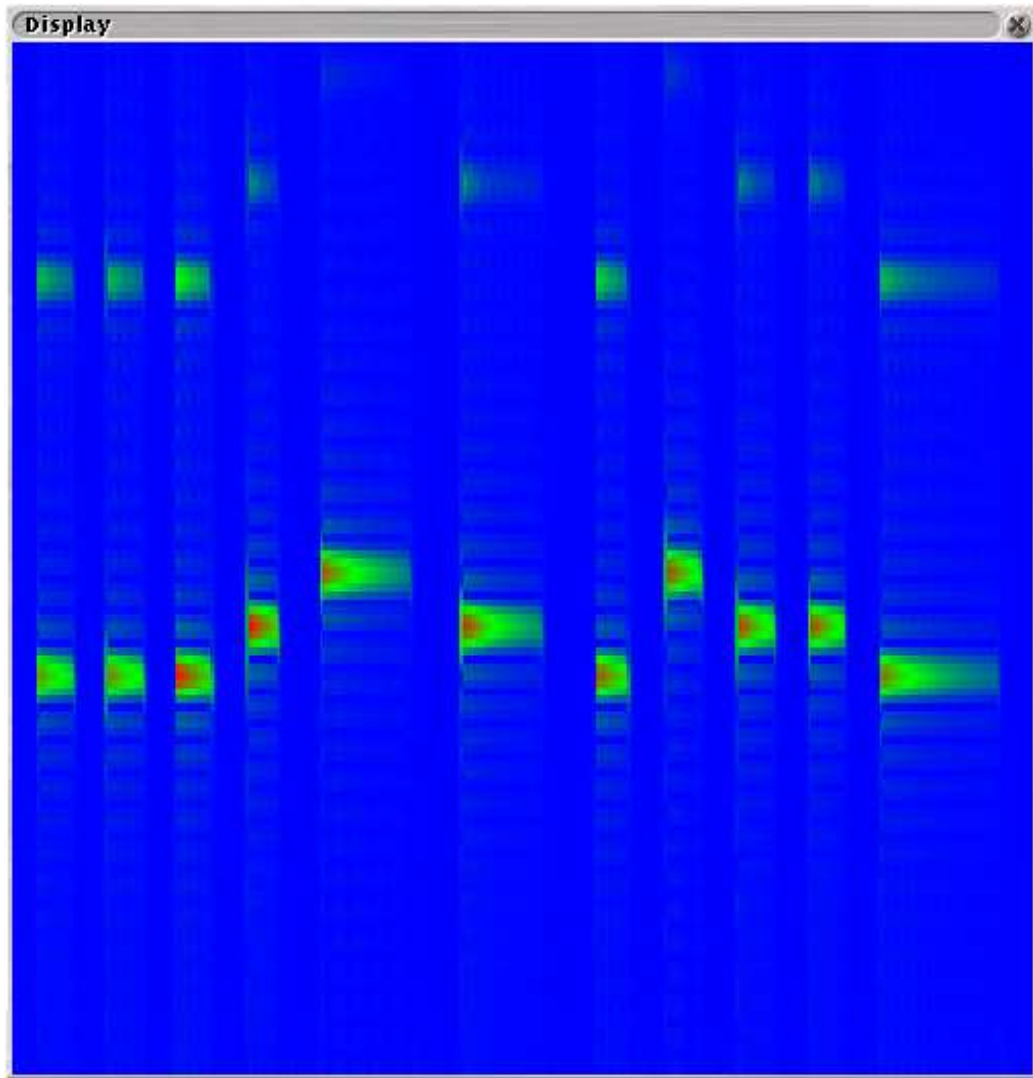


FIG. 10 – Ajout de zéros (facteur 8)

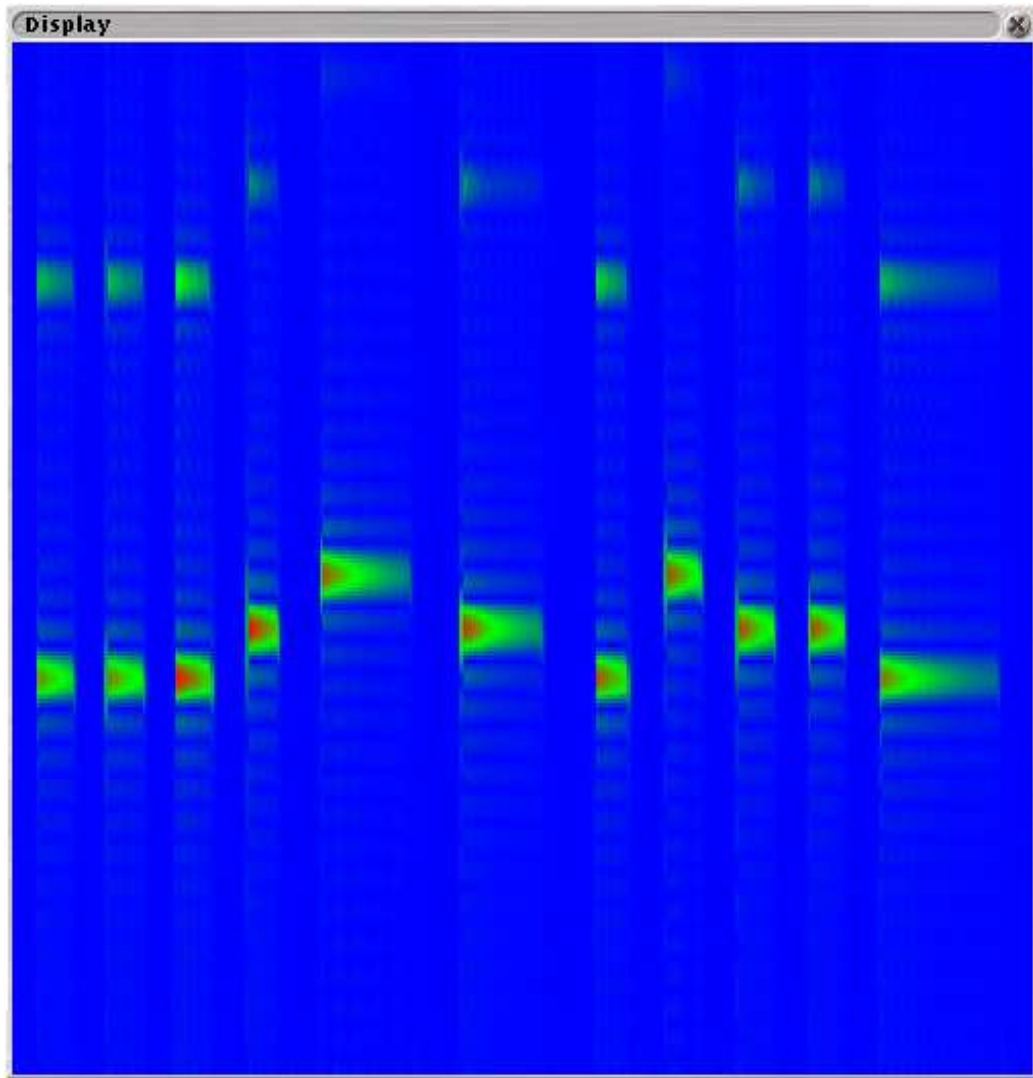


FIG. 11 – Ajout de zéros (facteur 16)

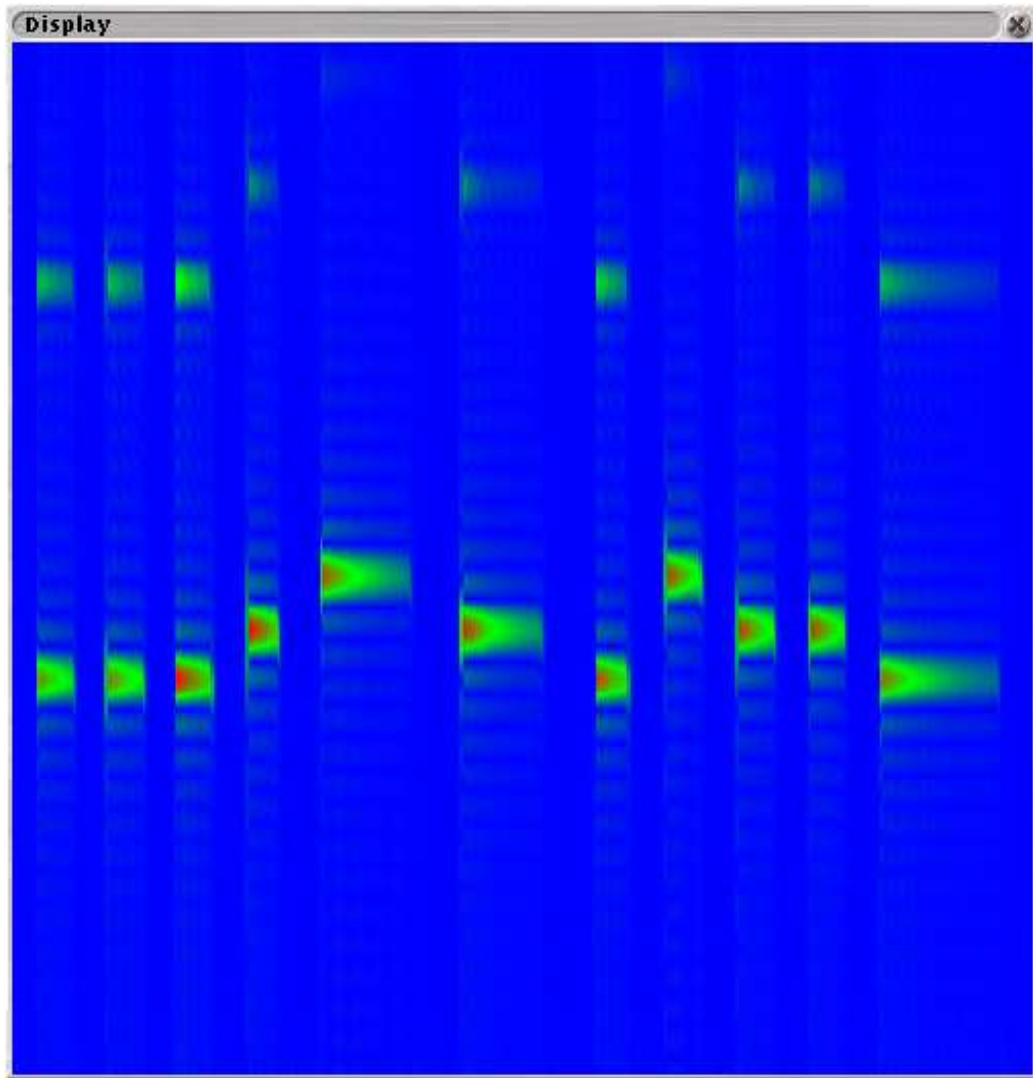


FIG. 12 – Ajout de zéros (facteur 64)

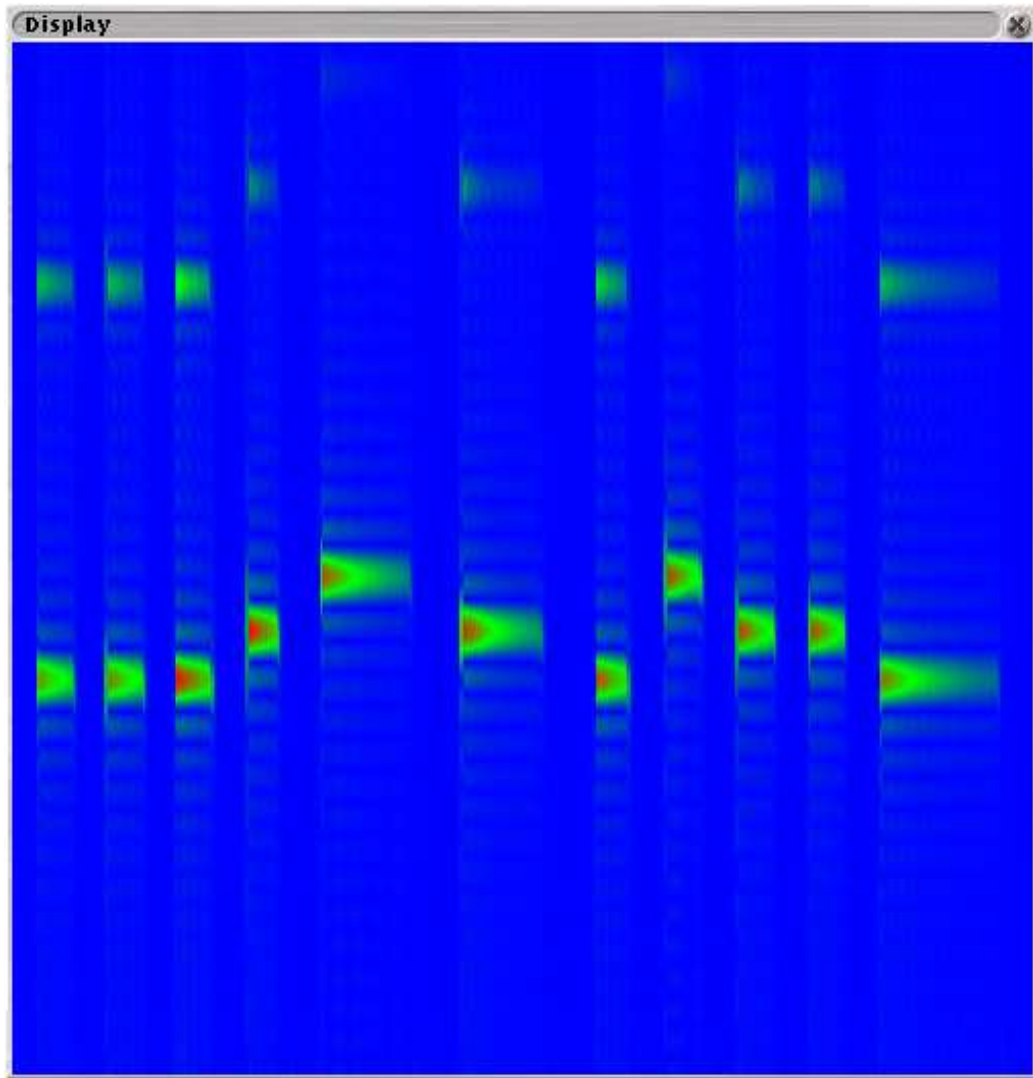


FIG. 13 – Problème d'harmoniques

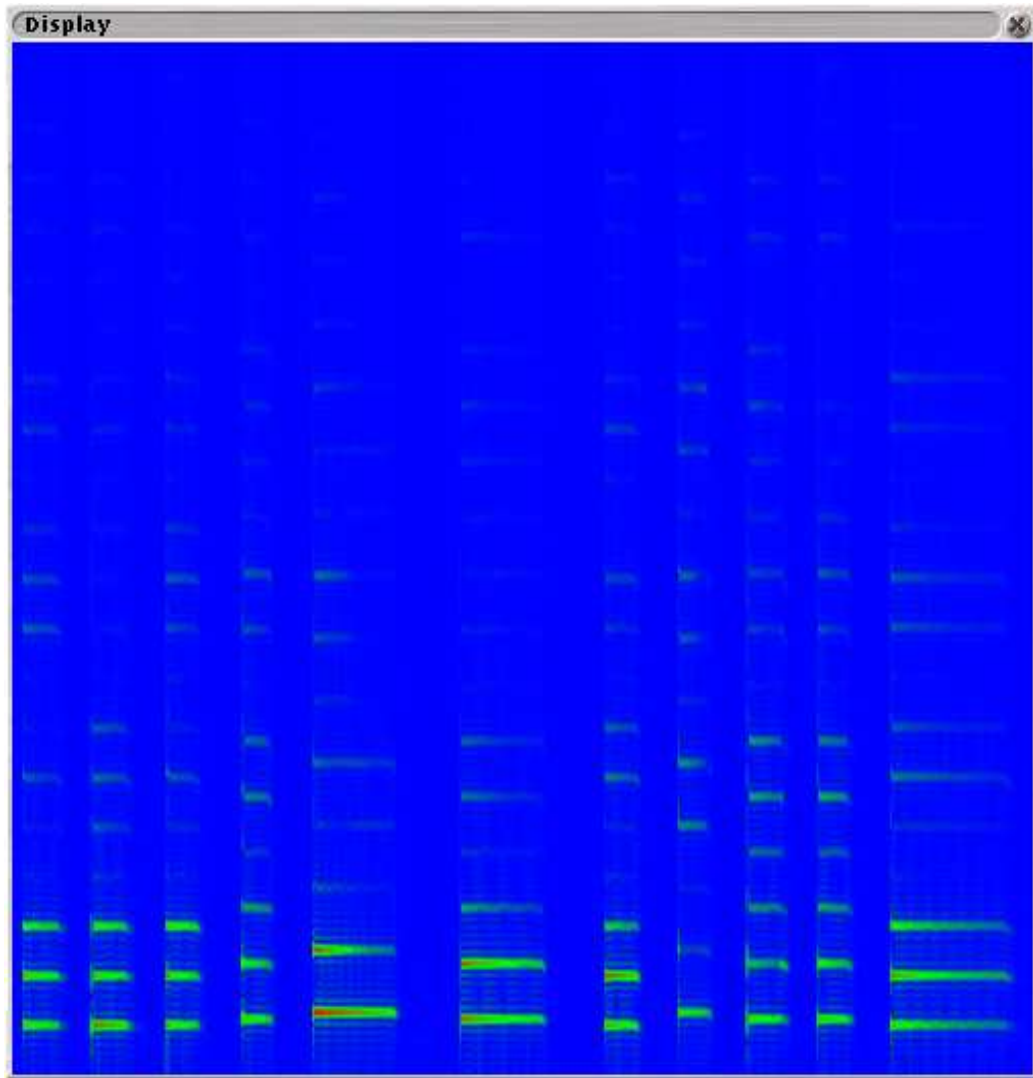


FIG. 14 – Signal à 8 Hz échantillonné à 100 Hz pendant 1 seconde

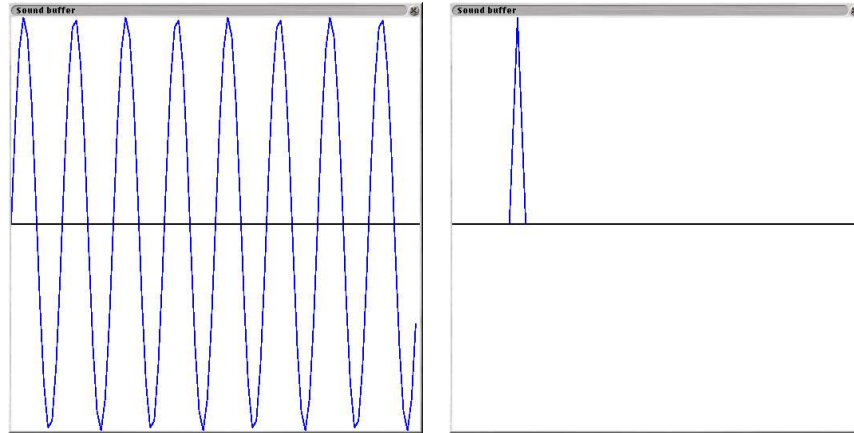


FIG. 15 – Signal à 8.5 Hz échantillonné à 100 Hz pendant 1 seconde

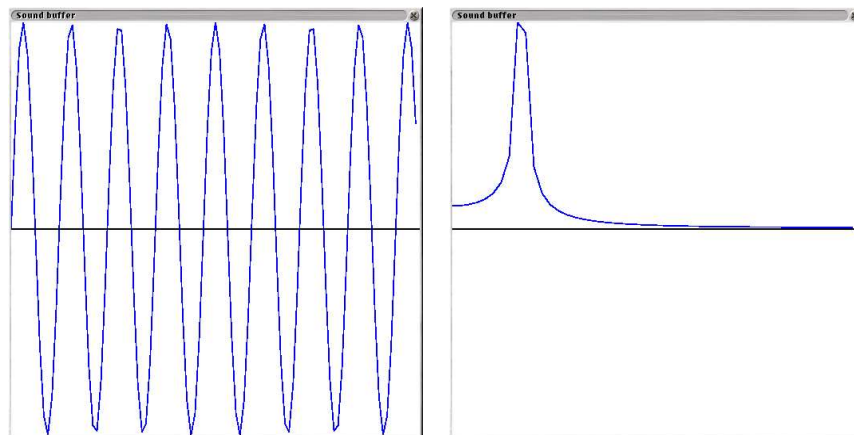
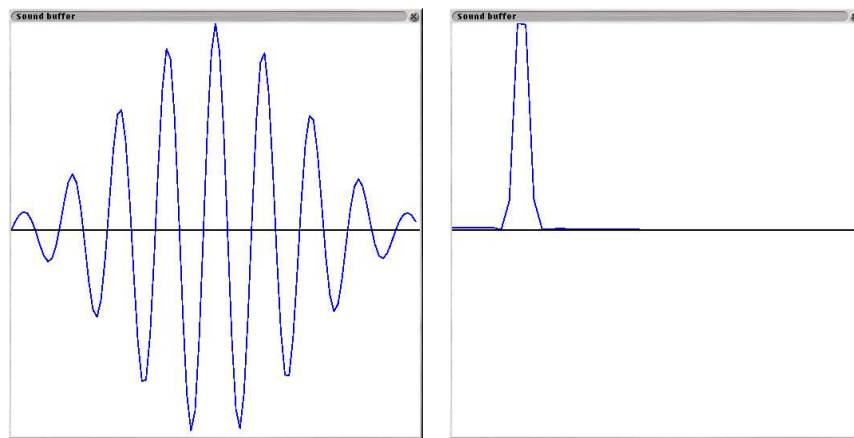


FIG. 16 – Signal à 8.5 Hz avec fenêtre de Hamming



5 Discussion

Les différentes méthodes et techniques qui ont été essayées ont produit des résultats intéressants. Une certaine discussion, qui fait l'objet du présent chapitre, s'impose afin de dégager des conclusions pertinentes de ces résultats. Tout d'abord, les résultats obtenus seront comparés avec ceux de produits équivalents. Ensuite, des conclusions plus générales, c'est-à-dire pertinentes pour la continuation de ce projet ou applicables à d'autres problèmes de traitement de signaux numériques, seront tirées des résultats obtenus. Finalement, les points qui n'ont malheureusement pu être abordés dans le cadre de ce travail seront donnés comme piste de travaux subséquents.

5.1 Autres outils de transcription automatique

Il peut être intéressant, afin d'analyser les résultats obtenus, de comparer avec d'autres outils de transcription automatique. On a choisi d'effectuer ce survol des produits existants après la majeure partie du travail afin de justement avoir certains résultats à comparer avec ceux de produits disponibles. Bien sûr, la nature exploratoire de ce projet incite à rester modeste quant aux prétentions du système développé à se comparer à d'autres systèmes ayant eu droit à beaucoup plus de travail.

5.1.1 SONIC

SONIC ([17]) est un système de transcription automatique de pièces polyphoniques de piano qui a pour sortie un fichier MIDI. Il est basé sur des réseaux d'oscillateurs adaptatifs qui permettent de détecter les différentes notes du piano ([14] et [15]).

Bien qu'il ait été conçu pour la transcription de piano, deux pièces jouées à la guitare, soit *Au Clair de la Lune* et quelques mesures de *Rondo Alla Turca* de Mozart (voir Annexe E), ont été données comme entrées à SONIC.

Les partitions correspondant aux fichiers MIDI produits par SONIC sont représentées dans les figures 17 et 18. Le rythme indiqué par la partition est plus ou moins significatif, car il a été construit par le logiciel de présentation. On voit toutefois que l'extraction de structures rythmiques est un des défis de la transcription automatique. Au niveau des notes, la transcription est

FIG. 17 – Résultat de SONIC pour Au Clair de la Lune

Musical notation for Au Clair de la Lune. The first staff, labeled '1', shows a melody in 4/4 time with a treble clef. The second staff, labeled '3', shows a bass line in 3/4 time with a treble clef. The notation includes various note values, rests, and accidentals.

FIG. 18 – Résultat de SONIC pour Rondo Alla Turca

Musical notation for Rondo Alla Turca, consisting of six staves labeled 1 through 6. The notation is complex, featuring multiple accidentals, triplets, and various rhythmic patterns. The first staff (1) is in 4/4 time with a treble clef. The second staff (2) is in 7/8 time with a treble clef. The third staff (3) is in 3/4 time with a treble clef. The fourth staff (4) is in 4/4 time with a treble clef. The fifth staff (5) is in 4/4 time with a treble clef. The sixth staff (6) is in 4/4 time with a treble clef. The notation includes various note values, rests, and accidentals.

généralement bonne, à l'exception d'un assez grand nombre de notes superflues. Ces notes sont habituellement situées un ou deux octaves au-dessus de la véritable note, ce qui indique que le système a pris une harmonique de la note fondamentale pour une seconde note. Le fait que la guitare et le piano n'ont pas le même patron d'intensité des modes de vibration secondaires des cordes peut expliquer en partie ce phénomène.

Si le projet développé dans le cadre de ce travail avait tenu compte de la polyphonie, on aurait dû gérer ce problème de détection d'harmoniques comme étant de nouvelles notes. Comme le prototype ici développé ne considérait qu'une seule note à la fois, le problème ne se présentait pas.

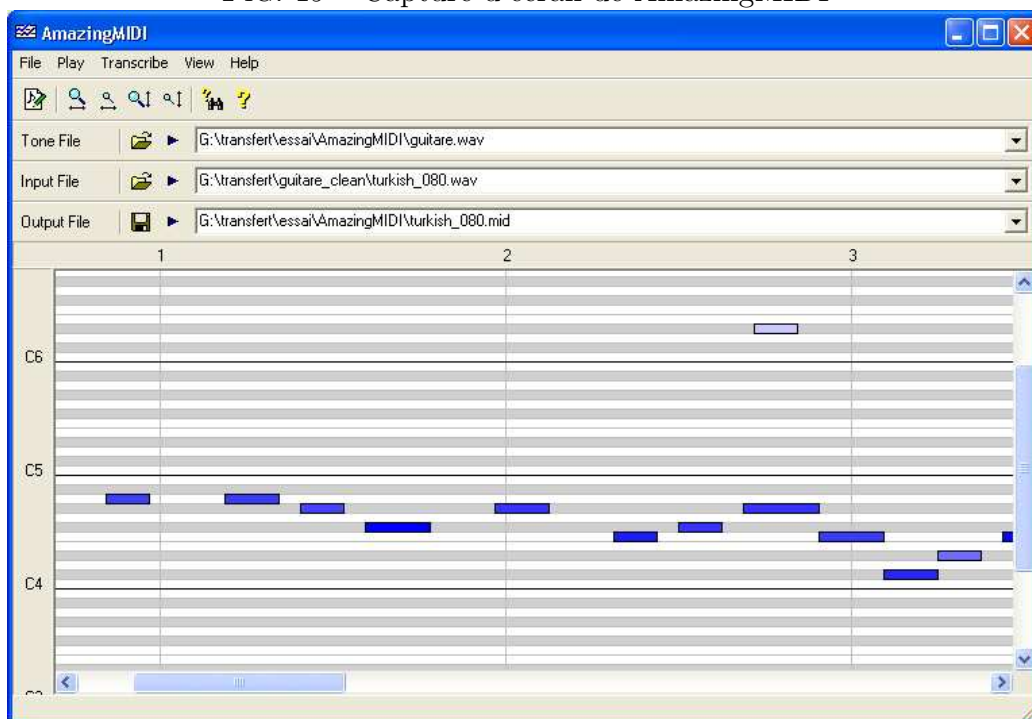
5.1.2 AmazingMIDI

Plusieurs points sont très intéressants à propos de AmazingMIDI ([2]). Tout d'abord, il s'agit d'un logiciel gratuit, bien qu'il ne soit pas *Open Source*. Ensuite, il prend en entrée un fichier nommé *Tone File* qui n'est qu'une brève note de l'instrument à transcrire, afin d'orienter le système dans sa reconnaissance de l'instrument. Ce système serait donc adaptable à plusieurs types d'instruments.

Finalement, le système présente une interface très intéressante, comme on peut le voir à la figure 19. En effet, on montre chacune des notes comme un rectangle commençant et terminant au début et à la fin de la note, à une hauteur correspondant à la note en question. Cette notation permet non seulement d'avoir en un seul coup d'oeil une vue d'ensemble du résultat de la traduction, ce qui en fait une représentation très intéressante pour aider le développement d'algorithmes de transcription¹⁸, mais elle constitue également une interface potentielle à l'édition de la transcription. Les transcriptions automatiques n'étant pas près d'être parfaites, une édition manuelle du résultat de la transcription risque d'être souvent nécessaire. Travailler sur cette série de rectangles (les étirer, les rapetisser, les déplacer, en effacer, en ajouter, etc.) représentés dans un plan, dont les deux échelles (le temps

¹⁸Cette représentation aurait pu être pertinente pour le développement de ce projet. Elle aurait constitué une représentation intermédiaire entre la liste exhaustive des notes et le son régénéré à partir de cette dernière, ce qui aurait permis de donner rapidement une appréciation qualitative de la justesse d'une transcription un peu complexe comme celle de la partie de Rondo Alla Turca utilisée pour tester le système.

FIG. 19 – Capture d'écran de AmazingMIDI



sur les abscisses et la hauteur de la note sur les ordonnées, comme avec une portée) sont évidentes, sera instinctif et efficace.

Ainsi, il aurait pu être intéressant d'ajouter une telle représentation de la liste de notes pour le programme développé dans le cadre de ce projet de fin d'études, bien qu'elle soit moins pertinente pour une transcription monophonique.

5.1.3 Autres

De nombreux autres produits de transcription, pas toujours automatique, de musique sont énumérés dans [20]. Quelques-uns d'entre eux ont été testés, sans donner de résultats ressortant assez des autres pour que ce soit pertinent de les mentionner. De nombreux produits commerciaux (comme [1] ou [19]) ont une version d'évaluation limitée qui ne permet pas de sauvegarder la sortie de la transcription, souvent en MIDI, ce qui complique l'analyse des

résultats. Le lecteur intéressé est invité à consulter [20] pour une excellente liste de produits.

5.2 Conclusions générales

Ce projet de fin d'études a permis d'en apprendre beaucoup sur le problème de transcription automatique de musique et sur le traitement de signaux numériques en général. On peut maintenant dégager certains faits importants à souligner et à retenir.

5.2.1 Nécessité d'abstraction

L'analyse d'un signal sonore numérique n'est pas simple, car la valeur des échantillons qui composent le signal n'a pas de signification en soi, mais uniquement lorsqu'elle est mise en relation avec les autres valeurs du signal. Contrairement au cas d'une image par exemple, où la valeur d'un pixel a une signification (couleur) autant que ses relations avec les pixels avoisinants (détection de contours), la valeur d'un échantillon sonore ne donne que bien peu d'information, ce qui complexifie le développement de systèmes d'analyse de signaux numériques.

En effet, il est nécessaire de travailler sur les signaux avec un certain degré d'abstraction, ou du moins à l'aide d'une représentation différente de la représentation temporelle des échantillons. Il est évident que cela complexifie la réparation de bogues, en plus de nécessiter un travail supplémentaire pour obtenir des informations pertinentes d'un signal. Par exemple, de nombreuses représentations graphiques des signaux, des résultats intermédiaires et des résultats finaux ont dû être implantées afin d'arriver à comprendre les résultats, autant en cas de succès que d'échec.

Ainsi, lorsque l'on travaille avec une représentation aussi peu significative pour l'humain que la représentation temporelle des échantillons d'un signal, on doit souvent fournir un effort supplémentaire pour interpréter les données, ce qui se traduit par un débogage plus difficile et un travail de développement d'« outils internes » souvent accru.

5.2.2 Résolution temporelle/fréquentielle

Une des premières réalités inévitables du traitement de signaux numériques est l'implacable opposition entre la résolution temporelle et la résolution fréquentielle lorsque l'on tente de séparer un signal numérique en une série de segments pour chacun desquels on calcule la transformée de Fourier. En effet, plus on prend de grands segments, plus il y aura d'échantillons dans un segment et donc plus le résultat de la transformée comportera de valeurs, ce qui divisera plus finement l'intervalle de fréquences allant de 0 à la moitié du taux d'échantillonnage. Ainsi, de grands segments donneront une résolution fréquentielle plus élevée.

Par contre, comme un plus grand nombre d'échantillons signifie une plus grande durée de temps pour chaque segment, un grand nombre d'échantillons par segment fait en sorte qu'il devient plus difficile d'associer l'analyse d'un segment à un moment précis, étant donné qu'un segment couvre une plus grande durée. Ainsi, la résolution temporelle s'en trouve affectée négativement.

En somme, la résolution temporelle diminue lorsque la résolution fréquentielle augmente, et vice-versa.

5.2.3 Bruit

En second lieu, le bruit est un facteur important à considérer pour la plupart des problèmes de traitement de signal. En effet, un niveau de bruit trop élevé peut amener un système à fournir des résultats complètement erronés, comme on a pu le voir avec les premiers échantillons enregistrés qui ont été soumis au système de TAM. On peut s'attaquer à un problème de bruit en :

- obtenant des signaux d'entrées avec moins de bruit : le bruit à l'intérieur d'un signal numérique provient souvent de son origine analogique, par exemple à cause de distorsions lors du transport du signal ou de sa conversion analogique-numérique par un microphone de piètre qualité. On peut tenter de minimiser les pertes à ces niveaux ;
- améliorant les algorithmes et heuristiques utilisés pour qu'ils tolèrent le bruit : cette solution peut être ardue, mais est parfois nécessaire. Il

s'agit toutefois d'un sujet trop complexe pour qu'il soit abordé ici.

Peu importe l'approche choisie pour contrer le bruit, on doit pratiquement toujours tenir compte de ce facteur en analyse ou traitement de signaux numériques.

5.2.4 Techniques

Par ailleurs, l'utilisation de certaines techniques se retrouve dans de nombreuses solutions à des problèmes de traitement numérique de signaux.

Ce projet de fin d'études a entre autres utilisé l'ajout de zéros pour obtenir une subdivision plus fine du domaine des fréquences, ainsi que l'application de la fenêtre de Hamming pour réduire l'apparence de certains artéfacts. Ces techniques, qui ont été très efficaces pour la TAM, peuvent être utiles dans une multitude de problèmes et doivent donc être gardées à l'esprit. Toutefois, leur application n'est pas systématique et doit être faite avec soin. Par exemple, bien que la fenêtre de Hamming ait convenu pour ce travail, un autre type de fenêtre pourrait être plus approprié pour une application différente. On doit donc user de jugement pour l'utilisation des techniques.

Ainsi, on a montré qu'il existe des techniques connexes à l'utilisation de la FFT qui permettent d'obtenir des résultats correspondant davantage aux besoins du problème, lorsqu'elles sont utilisées adéquatement.

5.3 Conclusions relatives à la TAM

Le travail effectué dans le cadre de ce projet a également permis d'identifier quelques points importants à considérer lorsque l'on s'attaque au problème de TAM.

5.3.1 Exponentialité de la croissance des fréquences

D'abord, une caractéristique importante de l'analyse de signaux sonores en particulier a été mise en évidence dans ce projet : l'échelle des fréquences perçues par l'oreille humaine évolue de manière exponentielle par rapport à la note perçue. L'importance de cette constatation est en partie due au fait que la FFT, algorithme qui a complètement révolutionné le monde du

traitement de signaux par les nouvelles possibilités créées par l'énorme gain d'efficacité qu'il procure, donne une représentation linéaire des fréquences qui composent un signal.

Cette sortie peut ne pas convenir à des analyses au niveau musical, car les notes que l'homme perçoit sont plus près les unes des autres dans la partie basse l'échelle des fréquences. La FFT donne également l'intensité d'un grand nombre de fréquences qui ne sont pas nécessairement pertinentes pour la solution du problème abordé, ce qui peut coûter cher en résolution temporelle.

Ainsi, l'exponentialité de la croissance des fréquences en fonction des notes perçues, par rapport à la linéarité du résultat de la FFT, soulève un certain dilemme entre un calcul performant et un autre mieux adapté aux besoins du problème.

5.3.2 Importance de l'interprétation

L'entrée d'un véritable système de transcription automatique est en fait le résultat d'une interprétation musicale effectuée par un humain. À l'instar d'un système qui décrypte une écriture manuscrite comportant des parties à peine lisibles, un système de TAM peut avoir de la difficulté à identifier certaines notes à cause de l'interprétation comme telle.

En effet, lors du travail d'analyse des résultats de transcription produits par le système développé dans le cadre de ce projet ainsi que ceux d'autres outils de TAM, certaines notes des pièces utilisées ont causé des problèmes à pratiquement tous les systèmes de TAM. Une note un peu manquée, jouée un peu moins forte, avec une attaque un peu moins franche, est plus difficile à transcrire, mais fait pourtant partie de la réalité des musiciens.

Cela porte à réaliser l'importance de l'interprétation qui est donnée en entrée à un système de TAM. Les notes ratées ou étouffées entraînent presque toujours une mauvaise transcription de la part du système de TAM, ce qui incite au souci de la qualité de la pièce transcrite automatiquement, ou alors exige un effort nouveau dans les outils de TAM pour régler ce genre de problèmes.

5.3.3 Utopie d'une solution parfaite

Le point précédent peut être généralisé d'une manière quelque peu pessimiste : il sera probablement toujours possible d'améliorer la qualité de la transcription automatique, mais l'atteinte d'un niveau de transcription parfaite pour toutes les entrées est assez utopique. Effectivement, malgré les progrès constants des différentes techniques de TAM, il existera toujours des cas complexes, difficilement interprétables, etc., qui feront que les transcriptions automatiques pourront toujours comporter certaines erreurs.

Cela dit, l'effort de recherche visant à améliorer les techniques courantes de TAM et à en développer de nouvelles n'est pas vain, car il y a encore beaucoup de place pour l'amélioration. Par contre, étant donné que la perfection des transcriptions est un objectif difficile, voire impossible, à atteindre, il sera toujours nécessaire de travailler les résultats d'une transcription automatique. Un autre côté du problème nécessitera donc des efforts.

En effet, si on veut arriver à produire des outils de TAM utiles, un effort doit être mis en oeuvre pour pouvoir manipuler le résultat de la transcription automatique pour en obtenir des données intéressantes et utilisables. Si on revient à l'exemple d'un système qui lit l'écriture manuscrite, les erreurs peuvent facilement être corrigées à l'aide de n'importe quel outil de traitement de texte. De la même façon, il devra être possible d'ajouter, d'effacer ou de corriger les notes produites par la TAM. Il est évidemment intéressant d'arriver à réduire de moitié le nombre de notes absentes reconnues quand même, mais il pourrait s'avérer plus utile de réduire de moitié le temps nécessaire à l'utilisateur pour effacer une note, ce qui procurerait presque le même gain, en plus d'être également applicable aux notes qu'il faudra inévitablement retirer.

Les instruments « analogiques », comme la guitare, la trompette ou même la voix, deviendraient donc de nouveaux périphériques d'entrée de données pour l'ordinateur et s'utiliseraient en complément des périphériques habituels tels que le clavier et la souris pour arriver, par exemple, à la rédaction de partitions la plus efficace et la plus agréable possible. Toutefois, très peu de travaux ([16] en est un) se consacrent à l'étude d'interfaces utilisant les instruments de musique.

En somme, une des conclusions les plus importantes de ce projet de fin d'études sur l'avenir des outils de TAM est que le plus grand pas à faire pour une utilisation réelle de ces derniers touche la manière dont ils sont présentés aux utilisateurs ainsi que la façon de les utiliser dans le cadre d'une tâche réelle.

5.4 Poursuite des travaux

Ce projet de fin d'études a constitué un contexte intéressant pour approcher le problème de TAM, bien que le cadre de travail était trop restreint pour tenter le résoudre d'une manière totalement efficace. En effet, il aurait été irréaliste de penser dépasser le travail de produits commerciaux à l'aide de ce projet de fin d'études de 135 heures, qui représentent moins d'un mois de travail d'un employé à temps plein.

Le temps a effectivement manqué pour la partie « développement de solutions » comme telle, avec tout le travail connexe qui a été effectué dans le cadre de ce projet : apprivoisement des outils, développement d'un cadre de travail de base, enregistrement de pièces, essai de produits, rédaction du rapport, etc. Toutefois, les objectifs du projet ont été atteints, car une bonne base a été jetée pour des travaux futurs auxquels l'auteur pourrait vouloir se consacrer.

En effet, il pourrait être extrêmement intéressant de se lancer dans l'automatisation de la transcription de pièces potentiellement polyphoniques. Beaucoup des algorithmes et heuristiques utilisés dans ce projet seraient relativement adaptables à la polyphonie. Toutefois, ces derniers transcrivent déjà avec assez de difficulté les pièces qui leur sont données qu'il faudrait les rendre plus robustes avant de les soumettre à la transcription de la polyphonie, qui ajoute de nombreux problèmes notamment l'interférence que les notes jouées simultanément peuvent exercer les unes sur les autres.

Par ailleurs, l'analyse d'effets spécifiques à la guitare, comme le « bend », est un des points qui n'ont pas été couverts par le présent projet. La détection et l'identification de ce type d'effet, puis l'analyse particulière à la région où l'on a détecté un tel effet qui sera probablement nécessaire pour la mesure des paramètres (note de départ, note d'arrivée, durée de la montée, etc.)

constituent certes des sujets qui gagneraient à être approfondis.

Ces idées de projets subséquents potentiels ne constituent que quelques-unes des possibilités à explorer à propos du problème complexe de la TAM, plus spécifiquement appliquée à la guitare.

6 Conclusion

En somme, ce projet de fin d'études a permis de s'attaquer au problème de la transcription automatique de musique interprétée à la guitare. Ce dernier a d'abord été défini selon le point de vue de l'auteur, puis une première ébauche de système a été développée. Ce prototype, capable d'abord de ne transcrire que des pièces extrêmement simples, a été amélioré à l'aide de diverses techniques couramment utilisées pour l'analyse de signaux numériques.

Le système final développé dans le cadre de ce travail était en mesure d'effectuer une transcription de qualité de pièces monophoniques générées plutôt qu'enregistrées. Il a également fourni des transcriptions acceptables de pièces plus difficiles à transcrire, la difficulté provenant de leur tempo plus rapide et du fait qu'elles étaient des enregistrements de guitare.

Le développement de ce système a permis d'atteindre les objectifs de ce projet de fin d'études. En effet, le problème de la transcription automatique a été attaqué sous plusieurs angles, quelques problèmes fréquents du traitement de signaux numériques ont pu être abordés, ainsi que les solutions courantes pour y remédier. Finalement, un système fonctionnel de transcription automatique a été développé.

Ainsi, ce travail ayant donné une bonne base de connaissances sur le sujet à l'auteur, ce dernier serait désormais prêt à poursuivre le travail à un niveau plus avancé. Ce projet de fin d'études constituera peut-être les fondations d'une recherche plus poussée, au niveau de la maîtrise par exemple, sur la transcription automatique de musique.

7 Bibliographie

Références

- [1] AKoff Sound Labs, « Wave to Midi - Music Recognition Software - Audio to Midi Conversion », AKoff Sound Labs. [En ligne]. Disponible : <http://www.akoff.com/index.html>. [Consulté 29 novembre 2006].
- [2] Arakisoftware, « AmazingMIDI - WAV to MIDI converter for music transcription », Arakisoftware. [En ligne]. Disponible : <http://www.pluto.dti.ne.jp/araki/amazingmidi/>. [Consulté le 29 novembre 2006].
- [3] J. P. Bello, G. Monti, M. Sandler, « Techniques for Automatic Music Transcription », International Symposium on Music Information Retrieval, 2000. [En ligne]. Disponible : http://ciir.cs.umass.edu/music2000/papers/bello_paper.pdf. [Consulté 28 novembre 2006].
- [4] S. G. Blackburn, « Content Based Retrieval and Navigation of Music Using Pitch Contours », Thèse de doctorat, University of Southampton, Southampton, 2000. [En ligne] Disponible : <http://www.beeka.org/research/pubs/phdthesis.pdf>. [Consulté 19 novembre 2006].
- [5] A. de Cheveigné, H. Kawahara, « Multiple period estimation and pitch perception model », Speech Communication 27, pp. 175-185, 1999. [En ligne]. Disponible : <http://www.ircam.fr/equipes/pcm/cheveign/sh/ps/SpeechComm.ps>. [Consulté 28 novembre 2006].
- [6] R. Boulanger, « cSounds.com - Official Csound Website », R. Boulanger. [En ligne]. Disponible : <http://www.csounds.com/>. [Consulté le 24 novembre 2006].
- [7] W. Data, « My Favorite Classics - An Educational Piano Album, Level Three », Torrance, California : Chas. H. Hansen Music Corp., 1969.
- [8] E. De Castro Lopo, « libsndfile », E. De Castro Lopo. [En ligne]. Disponible : <http://www.mega-nerd.com/libsndfile/>. [Consulté le 24 novembre 2006].
- [9] M. Frigo, S. G. Johnson, « FFTW Home Page », MIT. [En ligne]. Disponible : <http://www.fftw.org/>. [Consulté le 24 novembre 2006].

- [10] A. P. Klapuri, « Automatic Transcription of Music », Proceedings of the Stockholm Music Acoustics Conference, August 6-9, 2003. [En ligne]. Disponible : http://www.cs.tut.fi/sgn/arg/klap/smac2003_klapuri.pdf. [Consulté 28 novembre 2006].
- [11] A. P. Klapuri, « Multiple fundamental frequency estimation based on harmonicity and spectral smoothness », IEEE Transactions on Speech and Audio Processing, Vol. 10, No. 6, pp. 804-816, November 2003. [En ligne]. Disponible : <http://www.cs.tut.fi/sgn/arg/klap/multiplef0.pdf>. [Consulté 28 novembre 2006].
- [12] A. P. Klapuri, « Signal processing methods for the automatic transcription of music », Thèse de doctorat, Tampere University of Technology, Tampere, 2004. [En ligne]. Disponible : <http://www.cs.tut.fi/sgn/arg/klap/phd/klap-phd.pdf>. [Consulté 28 novembre 2006].
- [13] V. Lazzarini, « The SndObj Sound Object Library Page », V. Lazzarini. [En ligne]. Disponible : <http://music.nuim.ie//musictec/SndObj/main.html>. [Consulté le 24 novembre 2006].
- [14] M. Marolt, « SONIC : transcription of polyphonic piano music with neural networks », Proceedings of Workshop on Current Research Directions in Computer Music, Barcelona, Spain, November 15-17, 2001. [En ligne]. Disponible : http://lgm.fri.uni-lj.si/matic/clanki/mosart_barcelona_2001_p.pdf. [Consulté 29 novembre 2006].
- [15] M. Marolt, M. Privosnik, « SONIC : a system for transcription of piano music », Dans : Kluev, V., D'Attelis, C. E., Mastorakis, N. E. (eds.), Advances in automation, multimedia and video systems and modern computer science, WSES Press, cop. 2001. [En ligne]. Disponible : <http://lgm.fri.uni-lj.si/matic/clanki/malta2001.pdf>. [Consulté 29 novembre 2006].
- [16] F. Mohamed, S. Fels, « KEYed user interface : Tools for expressive music production », Proceedings of the International Computer Music Conference (ICMC), Gothenburg, Sweden, pp. 88-91. 2002. [En ligne]. Disponible : <http://hct.ece.ubc.ca/research/lmnkui/KEYED.pdf>. [Consulté 2 décembre 2006].
- [17] M. Marolt, « SONIC », Faculty of computer and information science, University of Ljubljana. [En ligne]. Disponible : <http://lgm.fri.uni-lj.si/sonic.html>. [Consulté 29 novembre 2006].

- [18] L. Rabiner, M. Cheng, A. Rosenberg, C. McGonegal, « A comparative performance study of several pitch detection algorithms », IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 24, No. 5, pp. 399-418, October 1976. [En ligne]. Disponible : <http://ieeexplore.ieee.org/iel6/29/26120/01162846.pdf>. [Consulté 28 novembre 2006].
- [19] G. Raskinis, « Wave to MIDI conversion and music transcription software - Solo Explorer », Recognisoft. [En ligne]. Disponible : <http://www.recognisoft.com/>. [Consulté 29 novembre].
- [20] Seventh String, « Transcription Resources », Seventh String. [En ligne]. Disponible : <http://www.seventhstring.com/resources/transcription.html>. [Consulté 29 novembre 2006]
- [21] S. W. Smith, « The Scientist and Engineer's Guide to Digital Signal Processing (Second Edition) », San Diego, California : California Technical Publishing, 1999. [Livre électronique]. Disponible : <http://www.dspguide.com/>. [Consulté le 6 juin 2006].
- [22] D. Temperley, « The Cognition of Basic Musical Structures », Cambridge : MIT Press, 2001. 404 pp. ISBN 0-262-20134-8.
- [23] T. Tolonen, M. Karjalainen, « A computationally efficient multipitch analysis model », IEEE Transactions on Speech Audio Processing, Vol. 8, No. 6, pp. 708-716, November 2000. [En ligne]. Disponible : <http://ieeexplore.ieee.org/iel5/89/18967/00876309.pdf>. [Consulté 28 novembre 2006].
- [24] C. Weihs, U. Ligges, « Automatic Transcription of Singing Performances », Bulletin of the International Statistical Institute, 54th Session, Proceedings, Volume LX, Book 2, 507-510. 2003. [En ligne]. Disponible : <http://www.statistik.uni-dortmund.de/de/content/forschung/-publikationen/forschungsberichte/fb03-1.pdf>. [Consulté 28 novembre 2006].
- [25] C. Weihs, U. Ligges, « Parameter Optimization in Automatic Transcription of Music », Dans : From Data and Information Analysis to Knowledge Engineering, M. Spiliopoulou, R. Kruse, A. Nürnberger, C. Borgelt, W. Gaul (Hrsg.), Springer-Verlag, Berlin, 741-747. 2006. [En ligne]. Disponible : <http://www.sfb475.uni-dortmund.de/berichte/tr39-05.pdf>. [Consulté 28 novembre 2006].

8 Glossaire

Bend Terme anglophone désignant l'action de pousser sur une corde de guitare perpendiculairement à cette dernière afin d'en augmenter la tension et ainsi modifier le son produit lorsque la corde est jouée. Cet effet permet de passer d'une note à une autre de manière continue, plutôt que discrète, sur la guitare.

DFT *Discrete Fourier Transform*, ou transformée de Fourier discrète. Transformation qui permet d'exprimer un signal dans le domaine des fréquences (comme somme de sinusoides) à partir d'un signal exprimé dans le domaine du temps.

FFT *Fast Fourier Transform*, ou transformée rapide de Fourier. Algorithme très efficace permettant de calculer la transformée de Fourier discrète.

Harmonique Composante sinusoidale d'un son dont la fréquence est un multiple entier de la fréquence fondamentale de la note jouée. L'intensité relative des harmoniques est un des facteurs qui permettent de distinguer le son produit par différents instruments.

Intervalle musical Différence entre deux notes, du point de vue de la théorie musicale. Se compte en demi-tons ou à l'aide d'un intervalle nommé, comme la tierce majeure ou la quinte.

Métronome Appareil donnant le rythme auquel une pièce doit être jouée afin de guider le musicien qui l'interprète.

MIDI *Musical Instrument Digital Interface*. Protocole qui, entre autres, définit chaque note d'un instrument de musique d'une manière précise et concise, ce qui permet aux instruments électroniques et aux ordinateurs de communiquer ce genre d'information en temps réel.

Monophonique Se dit d'une mélodie où une seule note est jouée à la fois.

Polyphonique Se dit d'une mélodie où plusieurs notes peuvent être jouées à la fois.

Résolution fréquentielle Précision de la détermination de la fréquence à laquelle une note est jouée, ce qui permet d'identifier avec plus ou moins de précision ladite note.

Résolution temporelle Précision de la détermination du temps exact du début et de la fin d'une note.

TAM Transcription automatique de musique. Procédé par lequel une partition est produite automatiquement à partir d'un signal sonore numérique, reproduisant la tâche que ferait un musicien qui apprend une pièce à l'oreille.

Taux d'échantillonnage Traduction libre de *sampling rate*. Il s'agit du taux auquel un signal est échantillonné, c'est-à-dire le nombre d'échantillons par unité de temps qui décrit le signal numérique.

Trille Alternance très rapide d'une note avec celle qui lui est supérieure (ou inférieure) généralement d'un demi-ton.

Vibrato Petite oscillation rapide de la hauteur d'une note qui « vibre » au lieu d'être simplement tenue un certain temps.

9 Annexe A : Description du projet

Cette annexe contient le texte qui a été remis au département pour la description de ce projet de fin d'études.

Mon projet, qui sera plutôt un travail de recherche, consistera en le développement d'un programme de transcription automatique de musique. En d'autres termes, le programme devra pouvoir, à partir d'un signal sonore numérique (un son contenu dans un fichier .wav ou .mp3, l'entrée de la carte de son, etc.), identifier les notes qui composent la musique, comme le ferait un musicien qui apprend une mélodie à l'oreille. Étant un guitariste, la transcription automatique de guitare sera l'objet principal de ce projet. La réalisation de ce dernier se fera en trois étapes.

Tout d'abord, un certain travail de recherche sera fait afin d'avoir un aperçu de ce qui se fait déjà dans le domaine : articles scientifiques, produits commerciaux, etc. Aussi, il est évident que certaines notions théoriques, notamment les concepts de base du traitement de signaux numériques et la fameuse transformée de Fourier, devront être approfondies pour mener à terme ce projet.

Ensuite, un prototype du type « preuve de concept » sera développé. Il visera tout d'abord à arriver à transcrire une mélodie monophonique (une seule note à la fois) très lente, ce qui représente probablement un des cas d'utilisation les plus faciles. Après cette étape, qui comprendra une bonne dose d'expérimentation des différentes idées et solutions potentielles au problème, les structures de données de base auront été développées. Elles se devront d'être à la fois performantes (bien que dans le cadre du projet de fin d'études, la transcription ne sera pas nécessairement faite en temps réel) et assez flexibles pour les modifications qui seront apportées à la troisième et dernière étape.

Finalement, la dernière étape consistera en l'amélioration du prototype selon le temps disponible. En effet, de nombreuses améliorations pourront être apportées :

- Le support des mélodies polyphoniques : être capable d'identifier deux ou plusieurs notes jouées simultanément ;*

- *L'amélioration de la résolution temporelle : pouvoir transcrire des mélodies très rapides, des trilles, etc. ;*
- *La transcription d'effets spécifiques à la guitare : « bend », « slide », « whammy bar », etc. ;*
- *La présentation des résultats dans un format standard : comme aucune spécification n'est faite à priori sur le format de sortie du résultat de la transcription, l'option la plus facile sera probablement choisie, c'est-à-dire une sortie à la console. Il pourrait toutefois être intéressant de produire la transcription dans un format plus standard, comme un fichier MIDI, une description de partitions en format Latex ou autre, etc. ;*
- *La création d'une interface usager graphique : la dernière chose sur la liste des priorités de ce projet de fin d'études est de produire un logiciel facilement utilisable par d'autres personnes que son développeur. Toutefois, il pourrait éventuellement être souhaitable de pouvoir travailler sur le résultat de la transcription afin de le modifier ou de le corriger en cas de besoin.*

Bien évidemment, les 135 heures allouées à ce projet de fin d'études seront bien insuffisantes pour réussir à réaliser toutes ces améliorations. Le maximum réalisable dans le temps disponible sera fait.

10 Annexe B : Décalage temporel

Cette annexe montre un exemple de décalage dans la production d'une partition. Voici la partition correcte pour le début de *Au Clair de la Lune* :



Toutefois, si le musicien interprétant cette pièce tient la première note de la deuxième mesure un peu trop longtemps, la partition transcrite pourrait ressembler à :



On voit donc que le résultat produit ne correspond pas à ce que l'on souhaiterait. Il est toutefois difficile pour une solution automatisée de gérer ce genre de problème.

11 Annexe C : CSound

Comme CSound([6]) occupe une place importante dans l'environnement de développement de ce projet, une brève explication s'impose.

CSound permet, à partir d'une description en format texte, de générer des sons. Les fichiers d'entrées de CSound sont composés de plusieurs sections, dont deux sont particulièrement intéressantes dans le cadre de ce projet.

La première section est la description des instruments qui serviront éventuellement à générer les notes d'une mélodie quelconque. Une fois les instruments définis, une autre section permet de « faire jouer » les instruments en spécifiant des notes ayant un temps de début, une durée, un instrument ainsi que d'autres paramètres spécifiques à l'instrument choisi (hauteur de la note, intensité, etc.).

CSound a donc été utilisé principalement pour deux raisons. Premièrement, il permettait de générer des fichiers audio d'entrée pour la transcription. Il était ainsi possible de contrôler totalement le signal et de ne pas être sujet aux différents problèmes inhérents à l'analyse d'une performance humaine. On a pu générer des pièces dont l'instrument pouvait être un signal sinusoïdal parfait, un signal où sont atténués le début et la fin de chaque note, etc. En second lieu, CSound s'est avéré un format de sortie extrêmement intéressant. En effet, il était possible d'entendre la transcription afin d'effectuer rapidement une première comparaison avec la pièce originale. De plus, comme la pièce est vraiment décrite comme une liste de notes avec leur temps de départ et leur durée, il était également facile de faire une analyse plus approfondie, étant donné la clarté et la simplicité du format texte de CSound.

12 Annexe D : Effet d'un décalage des valeurs d'un signal

Les premiers enregistrements de guitare ont été effectués en branchant directement la guitare dans l'entrée de la carte de son de l'ordinateur. Ces premiers enregistrements présentaient une particularité intéressante : le signal numérisé ne comportait que des valeurs négative, ce qui signifie qu'il n'était pas centré autour de 0, mais plutôt décalé d'un seul côté. Il a été possible ensuite de « normaliser » les signaux, c'est-à-dire corriger la valeur des échantillons pour que le signal soit centré autour de 0.

Toutefois, l'absence de cette correction revenait à prendre le signal « correct » (corrigé) et à ajouter (ou soustraire) une certaine valeur à tous les échantillons. Dans le domaine des fréquences, cette correction correspond à l'ajout d'un signal d'une fréquence de 0 (donc constant), ce qui a donné des résultats intéressants dans les représentations du domaine des fréquences.

La figure 22 montre le résultat de la transcription à l'aide de la première méthode, c'est-à-dire avec une distribution linéaire des fréquences. La bande rouge située dans les basses fréquences représente le bruit causé par le décalage par rapport à un signal centrée. On voit également qu'il est plus difficile de distinguer les notes dans le haut de la figure.

Il a été possible d'arriver à obtenir un résultat malgré cette distortion, en fixant un seuil en-deça duquel les fréquences étaient considérées comme du bruit et donc ignorées. Ce seuil a permis d'obtenir une transcription de qualité très comparable à celle de la table 3. Par contre, ce seuil a dû être fixé à une fréquence qui correspond à la note jouée lorsque la quatrième corde de la guitare est jouée à vide. Ainsi, si la mélodie comportait des notes situées sous ce seuil, elles auraient été tout simplement ignorées. Il est donc évident que ce seuil ne constitue pas une solution acceptable pour obtenir un système sérieux.

Le même problème de seuil se pose pour l'utilisation de la méthode logarithmique. Le bruit prend par contre une toute autre forme, comme on peut le voir sur la figure 23.

FIG. 22 – Transcription d'une pièce avec décalage

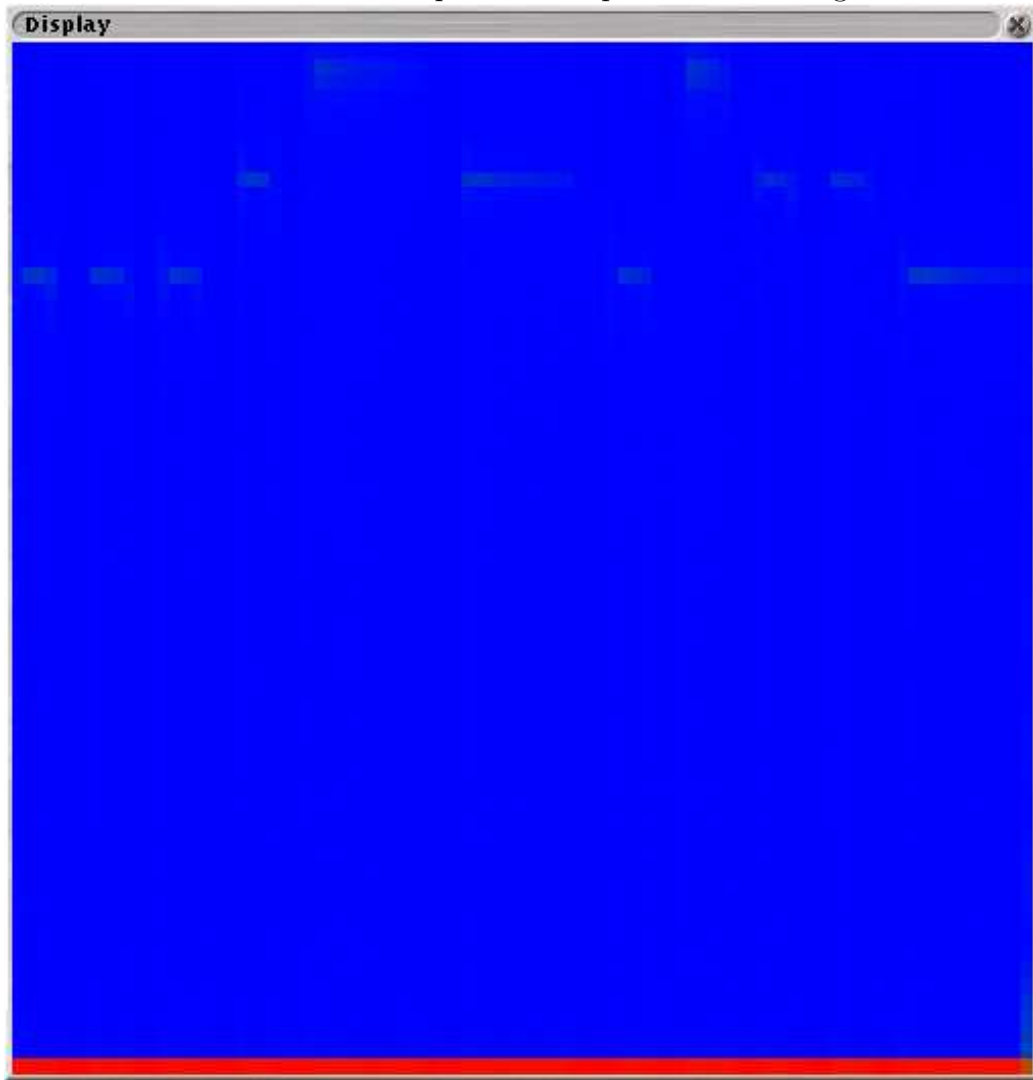
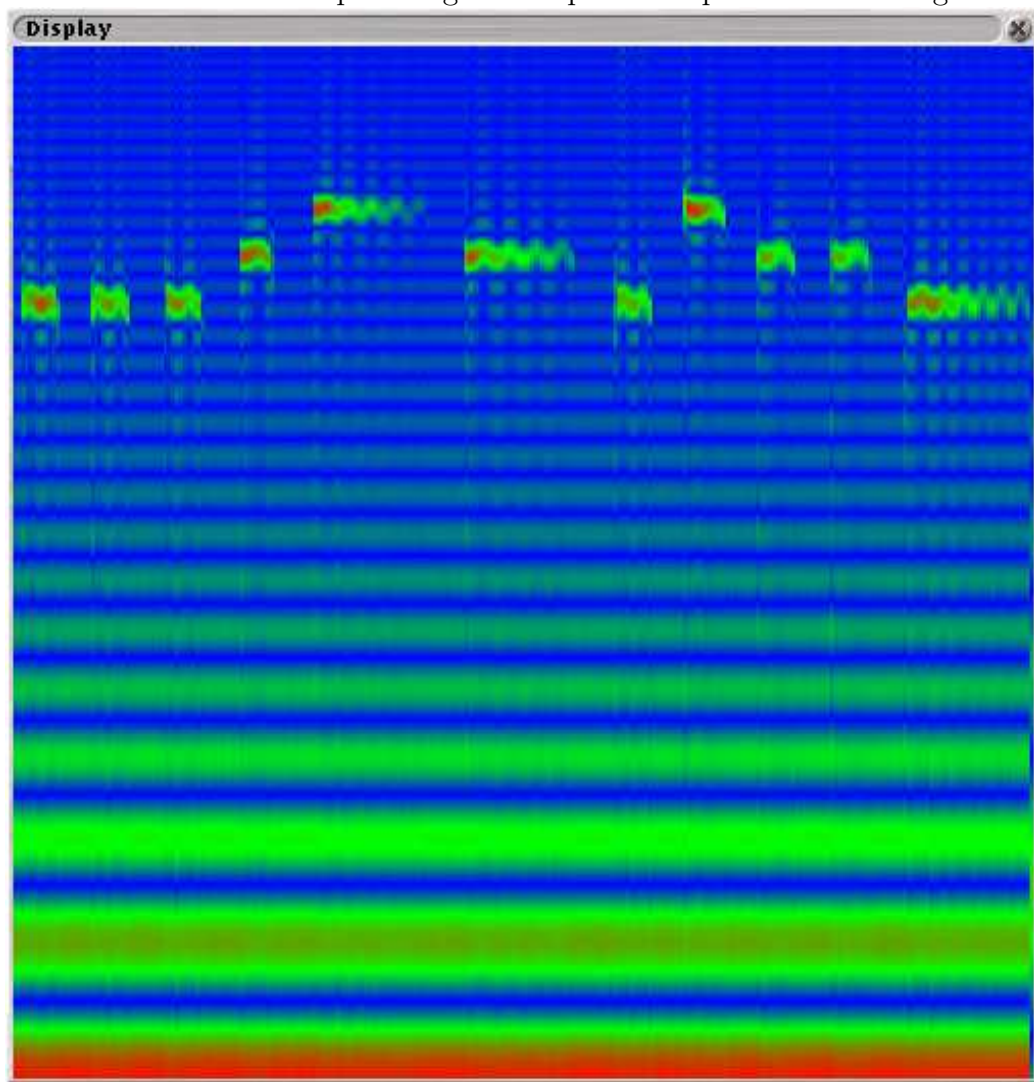


FIG. 23 – Transcription logarithmique d'une pièce avec décalage



13 Annexe E : Rondo Alla Turca

Voici les quelques mesures de la mélodie de Rondo Alla Turca, de Wolfgang Amadeus Mozart (tirée de [7]), qui ont été jouées à la guitare pour l'enregistrement d'un signal d'entrée au système. Seule la mélodie de la clé de sol a été jouée.

FIG. 24 – Rondo Alla Turca

The image displays a musical score for the Rondo Alla Turca by Wolfgang Amadeus Mozart. The score is written for a piano and includes two systems of music. The first system begins with the instruction *p leggiero*. The melody is written in the treble clef, and the bass line is in the bass clef. The key signature is G major (one sharp). The score features various fingerings and articulations, including slurs and accents. The second system concludes with a dynamic marking of *f* (forte).