République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique

Laboratoire Signal et Communications

Thèse de Doctorat D-LMD en Electronique

Option : Signal et Communications

Elaboration d'un Système de Synthèse par Sélection d'Unités en Vue de la Récitation du Saint Coran

Présentée par :

Mlle Nadjla BETTAYEB

Soutenue publiquement le **04 / 07 / 2021** devant le jury composé de :

Présidente :	Mme Rachida TOUHAMI	Professeur	ENP, Alger
Rapporteur :	Mme Mhania GUERTI	Professeur	ENP, Alger
Examinateurs :	Mme Nadjia BENBLIDIA	Professeur	USDB, Blida
	Mme Latifa HAMAMI	Professeur	ENP–ESDAT, Alger
	Mr Halim SAYOUD	Professeur	USTHB, Alger

ENP 2021

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique

Laboratoire Signal et Communications

Thèse de Doctorat D-LMD en Electronique

Option : Signal et Communications

Elaboration d'un Système de Synthèse par Sélection d'Unités en Vue de la Récitation du Saint Coran

Présentée par :

Mlle Nadjla BETTAYEB

Soutenue publiquement le **04 / 07 / 2021** devant le jury composé de :

Présidente :	Mme Rachida TOUHAMI	Professeur	ENP, Alger
Rapporteur :	Mme Mhania GUERTI	Professeur	ENP, Alger
Examinateurs :	Mme Nadjia BENBLIDIA	Professeur	USDB, Blida
	Mme Latifa HAMAMI	Professeur	ENP–ESDAT, Alger
	Mr Halim SAYOUD	Professeur	USTHB, Alger

ENP 2021

ملخص:
يوضح هذا العمل كيفية إنشاء جهاز لتركيب الكلام باللغة العربية الفصحى لتلاوة القرآن الكريم. تعتبر طريقة انتقاء الوحدات إحدى الوسائل المستعملة لإنتاج الكلام آليا، اذ تعتمد على سلسلة وحدات صوتية طبيعية بعد أن يتم اختيار أفضلها من قاعدة بيانات كبيرة. ولهذا يعتمد أداء هذه الطريقة على ثراء قاعدة البيانات وفاعلية خوارزمية الانتقاء. من اجل الوصول الى هدفنا قمنا ببناء قاعدة بيانات تضم 11077 وحدة صوتية مع خصائصها البروزودية والسياقية المختلفة. تبدأ عملية تركيب الكلام بالتحويل الحرفي-الصوتي للنص. وبعد ذلك تمّت تجزئة آلية الإنتقاء إلى مرحلتين متتاليتين لتقليص الوقت المستغرق للعملية. الأولى هي مرحلة الإختيار السياقي للوحدات عن طريق استمثال دالة تكلفة الهدف. أما الثانية، فهي إختيار يتم فيه تخفيض دالة لتكلفة التسلسل والبحث ذهابا وإيابا بالبرمجة الديناميكية. تتمثل مساهمتنا الرئيسية في اقتراح نهج جديد يعتمد على استعمال الأنظمة الخبيرة لضبط أوزان الخصائص المستخدمة في عملية الاستمثال. أظهرت الدراسة التقييمية جودة هذا الجهاز مع تلاوة صحيحة للقرآن. بالإضافة لذلك، قدّم الكلام المركب نسبة وضوح تقدر بـ 91.38٪ ومعدل طبيعي يبلغ 3.66 من 5.

الكلمات المفتاحية: تركيب الكلام اصطناعيا، انتقاء الوحدات الصوتية، اللغة العربية الفصحى، الأنظمة الخبيرة، القرآن الكريم.

Abstract:
This work presents the development of a standard Arabic Unit Selection Speech Synthesis (USSS) system for the Holy Quran recitation (HQ_TTS: Holy Quran Text-To-Speech). USSS is an automatic speech generation method that is based on the selection then the concatenation of natural sound segments, called units, from a large database. The performance of this method depends on the database richness and the selection algorithm efficiency. To achieve our goal, we first prepared a database consists of 11077 acoustic units with their distinct prosodic and contextual features. The HQ_TTS starts with a phonetic transcription of the text. After that, the selection algorithm is divided into two successive parts to optimize its speed. The first is the units contextual selection, which is based on the optimization of a target cost function. While the second is a selection that is done by minimizing a concatenation cost function and a forward-backward dynamic programing search. Our main contribution consists of proposing a new approach based on the use of Expert Systems to tune the parameters and features involved in this optimization. The evaluation results of the HQ_TTS system showed its performances with a correct recitation of the Quran. In addition, the synthesized speech resulted in an intelligibility of 91.38% and scored a naturalness rate of 3.66 from 5.

Key words: Speech Synthesis, Unit Selection, Standard Arabic, Expert System, Holy Quran.

Résumé :
Ce travail présente le développement d'un système de Synthèse par Sélection d'Unités acoustiques (SSU) de l'Arabe Standard (AS) pour la récitation du Saint Coran (HQ_TTS : Holy Quran Text-To-Speech). La SSU est une méthode de génération automatique de la parole qui se base sur la sélection puis, la concaténation des segments sonores naturels, appelés unités, à partir d'une grande Base de Données (BD). La performance de cette méthode dépend de la richesse de la BD et l'efficacité d'algorithme de sélection. Pour atteindre notre objectif, nous avons, d'abord, élaboré une BD constituée de 11077 unités acoustiques avec leurs caractéristiques prosodiques et contextuelles distinctes. Le HQ_TTS commence par une transcription phonétique du texte. Après cela, l'algorithme de sélection a été divisé en deux parties successives, afin d'optimiser la rapidité de ce système. La première est une sélection contextuelle des unités, qui se base sur l'optimisation d'une fonction de coût cible. Tandis que, la deuxième est une sélection qui se fait par minimisation d'une fonction de coût de concaténation et une recherche par programmation dynamique forward-bakward. Notre contribution principale consiste à proposer une nouvelle approche basée sur l'utilisation des Systèmes Experts (SE) pour ajuster les paramètres et les caractéristiques associés à cette optimisation. Les résultats d'évaluation du HQ_TTS ont montré sa performance avec une récitation correcte du Coran. La parole synthétisée donne une intelligibilité de 91.38 % et un score du naturel de 3.66 sur 5.

Mots clés : Synthèse de la Parole, Sélection d'Unités, Arabe Standard, Système Expert, Saint Coran.

Remerciement

Que Dieu soit loué pour nous avoir permis d'arriver au terme de ce travail.

Ce travail a été réalisé au sein du Laboratoire Signal et Communications (LSC), Département d'Electronique de l'École Nationale Polytechnique d'Alger (ENP). Il a été mené sous la direction de Professeur **GUERTI Mhania** que je tiens à remercier profondément pour son encadrement, son aide, ses directives, ses précieux conseils, ses critiques constructives et surtout pour sa disponibilité et sa compréhension.

Je tiens à remercier Mme **TOUHAMI Rachida** Professeur à l'ENP, pour m'avoir fait l'honneur de présider mon jury de thèse.

J'adresse aussi mes remerciements aux membres du jury, Mmes **BENBLIDIA Nadjia** professeur à l'USDB et **HAMAMI Latifa** professeur à l'ENP et l'ESDAT et Mr **SAYOUD Halim** Professeur à l'USTHB, pour avoir accepté d'examiner et juger ce travail.

Mes sincères remerciements vont à messieurs **ACHAB Nouredine** (que Dieu ait pitié de lui) et **GUERID Abdelkader** pour leurs aides et précieux conseils.

De plus, je remercie le professeur **Naeem Ramzan** de m'avoir invitée dans le cadre d'un stage à School of Engineering and Computing, University of West of Scotland (UWS), Royaume-Uni, pour son aide et conseils.

Je voudrais aussi remercier tous les enseignants du Département d'Electronique et tous les membres du LSC pour leurs encouragements et leur soutien.

Je remercie enfin tous les participants des tests d'évaluation pour le temps qu'ils avaient consacré.

Que tous ceux qui m'ont aidée, de près ou de loin, ne serait-ce qu'à travers leurs encouragements, trouvent ici l'expression de mes vifs remerciements .

Dédicaces

Je dédie cet humble travail à :

Mes chers Parents ;

Ma sœur et Mon frère ;

Mes amis et collègues ;

Je suis sincèrement reconnaissante à toute personne m'ayant aidée et soutenue durant la période de ce travail.

À vous tous, je suis fière de vous avoir.

نجلاء

Table des matières

Liste des tableaux

Liste des figures

Liste des abréviations

Introduction générale **15**

Chapitre 1 : Notions fondamentales sur la synthèse de la parole et la langue arabe **19**

1.1 Introduction 20
1.2 Généralités sur la parole 20
1.2.1 Niveau physiologique 20
1.2.2 Niveau acoustique 21
1.2.3 Niveau phonologique 22
1.2.4 Niveau phonétique 23
1.2.4.1 Phonétique auditive 23
1.2.4.2 Phonétique articulatoire 23
1.2.4.3 Phonétique acoustique 26
1.3 Langue arabe 26
1.3.1 Signes diacritiques 27
1.3.1.1 Caractéristiques phonologiques 28
1.4 Synthèse de la parole 29
1.4.1 Techniques d'analyse de la parole 30
1.4.1.1 Techniques temporelles 30
1.4.1.2 Techniques spectrales 32
1.4.2 Techniques de synthèse de la parole 37
1.4.2.1 Codage prédictif linéaire 37
1.4.2.2 Synthèse par formants 38

1.4.2.3 Synthèse articulatoire 38
1.4.2.4 Synthèse par l'algorithme PSOLA (Pitch Synchronous OverLap and Add) 39
1.4.3 Méthodes de synthèse de la parole 39
1.4.3.1 Synthèse paramétrique 39
1.4.3.2 Synthèse par concaténation 40
1.4.4 Systèmes de synthèse à partir du texte (Text-To-Speech : TTS) . . 42
1.4.4.1 Traitement linguistique 42
1.4.4.2 Traitement acoustique et prosodique 43
1.4.5 Défis et problèmes dans la synthèse de la parole 43
1.4.5.1 Dans le haut niveau 43
1.4.5.2 Dans le bas niveau 44
1.4.6 Evaluation des systèmes de synthèse de la parole 44
1.4.6.1 Méthodes d'évaluation 45
1.4.6.2 Evaluation de la qualité de la parole 46
1.5 Conclusion 48

Chapitre 2 : Synthèse par sélection d'unités, notions sur les systèmes experts et les algorithmes génétiques **49**
2.1 Introduction 50
2.2 Synthèse par Sélection d'Unités acoustiques 50
2.2.1 Algorithme de Black et Hunt 50
2.2.2 Base de données / Corpus 51
2.2.2.1 Corpus 51
2.2.2.2 Taille d'un corpus / BD 52
2.2.2.3 Qualité d'un corpus 53
2.2.2.4 Type des unités 53
2.2.3 Sélection des unités 54
2.2.3.1 Présélection 55
2.2.3.2 Fonction cible / coût cible 55
2.2.3.3 Fonction de concaténation / coût de concaténation . . 56
2.2.3.4 Pondération des coûts 57
2.2.3.5 Procédure de sélection 62
2.2.4 Concaténation des unités 64
2.3 Systèmes Experts (SE) 66

2.3.1 Architecture d'un système expert 66
2.3.1.1 Base des Connaissances (BCc) 67
2.3.1.2 Moteur d'Inférence (MI) 67
2.3.1.3 Interface d'Utilisation (IU) 68
2.3.1.4 Installation d'explications 68
2.3.1.5 L'acquisition de la connaissance 69
2.3.1.6 L'ingénieur des connaissances 69
2.4 Algorithme Génétique (AG) 69
2.4.1 Principe de fonctionnement 69
2.4.2 Eléments principales 70
2.4.2.1 Codage des données 71
2.4.2.2 Population initiale 71
2.4.2.3 Gestion des contraintes 72
2.4.2.4 Opérateur de croisement 72
2.4.2.5 Opérateur de mutation 72
2.4.2.6 Principes de sélection 73
2.5 Conclusion 74

Chapitre 3 : Elaboration du système HQ_TTS **76**
3.1 Introduction 77
3.2 Initiation sur le système HQ_TTS 77
3.3 Construction de la Base de Données (BD) 77
3.3.1 Préparation du corpus 78
3.3.2 Segmentation et annotation 78
3.4 Elaboration du HQ_TTS 82
3.4.1 Traitement du Langage Naturel (TLN) 83
3.4.1.1 Transcription Orthographique Phonétique (TOP) 83
3.4.1.2 Analyse du texte et segmentation en unités 88
3.4.2 Sélection des unités 89
3.4.2.1 Sélection par fonction cible (sélection contextuelle) . . . 89
3.4.2.2 Sélection acoustique (finale) 96
3.4.2.3 Sélection des unités pour les versets à lettres abrégées . 98
3.4.3 Concaténation des unités sélectionnées 99
3.5 Conclusion 99

Chapitre 4 : Evaluation du système HQ_TTS, résultats obtenus et discussions 100
4.1 Introduction 101
4.2 Performance des différents modules 101
4.2.1 Transcription orthographique phonétique (TOP) 101
4.2.2 Procédure de sélection 103
4.2.3 Ajustement des scores 105
4.3 Qualité de la parole 107
4.3.1 Intelligibilité 107
4.3.2 Naturel 111
4.4 Evaluations générales 114
4.5 Conclusion 118

Conclusions générales et perspectives 119

Références bibliographiques 122

Liste des tableaux

Tableau 1.1 Voisement, mode et lieu d'articulation des consonnes et semi-voyelles Arabe 25
Tableau 3.1 Code d'Alphabet Phonétique International (API) et code proposé des caractères arabes 81
Tableau 3.2 Transcription phonétique de quelques mots d'exception dans le dictionnaire 85
Tableau 3.3 Caractères et codes utilisés pour les règles de *tajweed* 88
Tableau 3.4 Valeurs des formants et de la fréquence fondamentale d'une voyelle « a » prononcée par un homme, extraites à l'aide du software speech analyzer 93
Tableau 4.1 Transcription manuelle de quelques phrases de test 102
Tableau 4.2 Comparaison des chaînes obtenues par les trois processus de recherche forward, bakward et combinée 105
Tableau 4.3 Pourcentage d'intelligibilité des phrases synthétisées 109
Tableau 4.4 Quelques mots synthétisés et leurs pourcentages d'intelligibilité 110
Tableau 4.5 Phrases utilisées dans le test de naturel de la parole et leurs pourcentages d'utilisation dans la BD 112
Tableau 4.6 Résultat du naturel de la parole synthétisée pour les phrases à choix libre 114
Tableau 4.7 Phrases de test de la consommation temporelle du système HQ_TTS 117
Tableau 4.8 Analyse de la consommation temporelle des fonctions du système HQ_TTS 118

Liste des Figures

Figure 1.1 Anatomie de l'appareil phonatoire humain 21
Figure 1.2 Plage d'audition de l'oreille humaine 23
Figure 1.3 Triangle vocalique de la langue arabe 24
Figure 1.4 Lieux d'articulation des phonèmes arabes 24
Figure 1.5 Audiogramme et spectrogramme du mot [bismi] qui présente quelques caractéristiques acoustiques des sons 26
Figure 1.6 Exemple des signes diacritiques de l'AS 27
Figure 1.7 Reconstruction de la machine de Kempelen par Wheatstone . . . 29
Figure 1.8 Diagramme de VODER 30
Figure 1.9 Spectrogrammes (a) à large bande et (b) à bande étroite 32
Figure 1.10 Diagramme général de calcul du vecteur caractéristique des valeurs MFCC et leurs dirivées 35
Figure 1.11 Spectre de la voyelle « aa » : (a) avant préaccentuation ; (b) après accentuation 35
Figure 1.12 Banc de filtres utilisé dans le calcul des MFCC 36
Figure 1.13 Schéma synoptique typique d'un synthétiseur statistique par HMM 40
Figure 1.14 Schéma synoptique d'un synthétiseur à partir du texte 43
Figure 2.1 Illustration de calcul des fonctions de coût 52
Figure 2.2 Exemple de modélisation en graphe du processus de sélection des unités 63
Figure 2.3 Structure des unités candidates dans la recherche de Viterbi . . . 64
Figure 2.4 Schémas de concaténation des diphones français [ʁe] et [ey] [23] 65
Figure 2.5 Eléments de composition d'un système expert 67
Figure 2.6 Principe des algorithmes génétiques 70
Figure 2.7 Croisement (a) à un point ; (b) à deux points 73
Figure 2.8 Principe de mutation 74
Figure 3.1 Segmentation de la phrase « [bismi llahi rraħmaani rraħiim] » . . 79

Figure 3.2 Exemple de segmentation d'une : (a) consonne fricative et une voyelle, (b) consonne occlusive 79
Figure 3.3 Exemple de segmentation de la *hamza* [ʔ] 80
Figure 3.4 Annotation des segments sonores extraits 80
Figure 3.5 Analyse par « Speech analyzer » et l'extraction des fichiers des caractéristiques acoustiques de type «.sft » 81
Figure 3.6 Partie d'un fichier « sft » contenant quelques caractéristiques acoustiques de l'unité «4aa222902 » 82
Figure 3.7 Schéma synoptique du système HQ_TTS 82
Figure 3.8 Etapes de la Transcription Orthographique Phonétique (TOP) dans le système HQ_TTS 84
Figure 3.9 Transcription et segmentation d'un verset ordinaire » 89
Figure 3.10 Transcription et segmentation d'un verset à lettres abrégées . . . 89
Figure 3.11 Schéma explicatif du processus d'entraînement des scores par AGai 92
Figure 3.12 Spectrogramme et audiogramme de la voyelle « a » précédée par : un phonème emphatique « x » (à gauche), phonème non-emphatique « f » (à droite) 93
Figure 3.13 Schéma synoptique du système expert (ES_US) 94
Figure 3.14 Exemple du processus d'affectation des scores par le moteur d'inférence 96
Figure 3.15 Recherche forward de la meilleure chaîne d'unités 97
Figure 3.16 Recherche backward de la meilleure chaîne d'unités 98
Figure 4.1 Résultat de la transcription automatique de quelques phrases de test 103
Figure 4.2 Temps de synthèse pour quelques phrases de test avec une et deux étapes de sélection 104
Figure 4.3 Nombre des unités candidates pour quelques phrases de test avec une et deux étapes de sélection 104
Figure 4.4 Résultat de comparaison de qualité de la parole de trois techniques de pondération des caractéristiques 106
Figure 4.5 Résultat de préférence de la qualité de la parole entre la synthèse avec : le SE et l'AGai 106

Figure 4.6 Nombre d'unités sélectionnées par la fonction cible en utilisant trois approches de pondération des scores 107
Figure 4.7 Boîte à moustaches des scores de test du naturel de la parole synthétique 113
Figure 4.8 Interface d'utilisation du système HQ_TTS 114
Figure 4.9 Profil d'exécution du système HQ_TTS 115
Figure 4.10 Profil d'exécution de la fonction « *interface* » du système HQ_TTS 116

Liste des abréviations

ACR	:	Absolute Category Rating
AG	:	Algorithmes Génétiques
AGai	:	Algorithme Génétique actif interactif
AGis	:	Algorithmes Génétiques interactifs
API	:	Alphabet Phonétique International
AS	:	Arabe Standard
BCc	:	Base de Connaissances
BD	:	Base de Données
CB	:	Chaîne Backward
CC	:	Chaîne Combinée
CCR	:	Comparison Category Rating
CELP	:	Codebook Excited Linear Prediction
CF	:	Chaîne Forward
DCR	:	Degradation Category Rating
DFW	:	Dynamic Frequency Wrapping
DMOS	:	Degradation Mean Openion Score
DRT	:	Diagnostic Rhyme Test
ES	:	Expert System
ES_US	:	Expert System for Unit Selection
FD-PSOLA	:	Frequency Domaine -Pitch Synchronous OverLap and Add
FFT	:	Fast Fourier Transform
GPS	:	Global Positioning System
HMM	:	Hidden Markov Model
HNM	:	Harmonic plus Noise Model
HPS	:	Psychoacoustic Sentences
HQ_TTS	:	Holy Quran Text-To-Speech
IA	:	Iintelligence Artificielle
IU	:	Interface d'Utilisaion
LPC	:	Linear Predictive Coding
LP-PSOLA	:	Linear Predictive Pitch Synchronous OverLap and Add
MBR-PSOLA	:	Multi-Band Resynthesis Pitch Synchronous OverLap Add

MFCC	:	Mel Frequency Cepstral Coefficients
MI	:	Moteur d'Inférence
MOS	:	Mean Opinion Score
MRT	:	Modified Rhyme Test
P / M	:	Phrase ou Mot
PESQ	:	Perceptual Evaluation of Speech Quality
PSOLA	:	Pitch Synchronous OverLap and Add
RAM	:	Random Access Memory
RAP	:	Reconnaissance Automatique de la Parole
RB	:	Recherche Backward
RC	:	Recherche Combinée
RCGA	:	Real Coded Genetic Algorithms
RELP	:	Residual Excited Linear Prediction
REP	:	Recherche dans l'Espace des Poids
RF	:	Recherche Forward
RML	:	Régression Multi-Linéaire
SC	:	Saint Coran
SE	:	Systèmes Experts
SNR	:	Signal to Noise Ratio
SNR_{seg}	:	Segmental Signal to Noise Ratio
SSU	:	Synthèse par Sélection d'Unités
SUS	:	Semantically Unpredictable Sentences
SVM	:	Support Vector Machine
TAP	:	Traitement Automatique de la Parole
TCD	:	Transformée en Cosinus Discrete
TD-PSOLA	:	Time Domaine Pitch Synchronous OverLap and Add
TF	:	Transformée de Fourier
TFD	:	Transformée de Fourier Discrète
TFDI	:	Transformée de Fourier Discrète Inverse
TLN	:	Traitement du Langage Naturel
TOP	:	Transcription Orthographique Phonétique
TS	:	Technique Standard
TTS	:	Text-To-Speech
USSS	:	Unit Selection Speech Synthesis
VODER	:	Voice Operation Demonstrator
ZCR	:	Zero-Crossing Rate

INTRODUCTION GÉNÉRALE

La synthèse automatique de la parole, ou TTS : Text-To-Speech, sert à générer une voix artificielle à partir d'un texte numérique. Elle est considérée comme une partie nécessaire dans les systèmes d'interaction Homme – Machine, qui sont devenus importants dans la vie courante. Un système TTS se compose, principalement, de deux grands modules, qui sont le Traitement du Langage Naturel (TLN) et le Traitement Automatique de la Parole (TAP). Le TLN sert à transformer le texte entré en une écriture phonétique. Tandis que, le TAP génère la parole à partir de cette représentation phonétique. Afin d'arriver à une parole synthétique proche de celle d'un être humain (intelligible, naturelle et expressive), plusieurs méthodes et techniques ont été appliquées, notamment dans le TAP. Parmi les méthodes adoptées actuellement, nous trouvons la Synthèse par Sélection d'Unités (SSU). Elle s'appelle aussi synthèse par corpus, où elle se base sur la sélection puis la concaténation des segments sonores naturels, appelés unités. Contrairement aux techniques classiques de concaténation (par diphone comme exemple), la SSU ne nécessite pas un traitement du signal vocal. Ceci par ce que les unités concaténées ont été bien sélectionnées à partir d'une large Base de Données (BD), de telle sorte que la parole résultante rapproche au maximum des caractéristiques du texte à synthétiser. De plus, elle offre, généralement, une parole synthétique plus naturelle que d'autres méthodes paramétriques ou statistiques, par ce qu'elle utilise des segments sonores réels. Comparant aux autres langues comme le Français ou l'Anglais, les synthétiseurs de parole arabe se sont répandus que pendant ces deux dernières décennies, et les recherches sont encore en progression pour les améliorer. La plupart de ces recherches se font dans le TLN, car c'est un grand défi dans la langue arabe à cause de l'absence des signes diacritiques dans les textes d'utilisation générale. Tandis que les travaux fait dans le TAP, concentre beaucoup sur la synthèse statistique paramétrique et la synthèse par concaténation de diphone, contrairement à la SSU qui est rarement utilisée. Cela peut être dû à la non disponibilité d'une BD personnalisée pour la synthèse de la langue arabe.

Le processus de sélection dans la SSU est, généralement, guidé par l'optimisation d'une fonction de coût. Cette dernière modélise d'une part le degré de correspondance entre les unités cibles, issues du texte, et celle de la BD, et d'autre part les discontinuités entre les unités consécutives à concaténer. Pour cela, de nombreuses recherches ont porté sur l'ajustement des paramètres et caractéristiques des unités impliquées dans ce processus. En particulier, la pondération de ces caractéristiques

qui est très importante dans la conception de la fonction de coût. C'est une tâche difficile, car elle doit refléter l'importance relative de chaque caractéristique utilisée pour sélectionner la chaîne d'unités la plus appropriée. Plusieurs approches ont été testées pour le problème d'ajustement des poids, dans lesquelles nous distinguons: l'ajustement manuel par un expert après quelques tests d'écoute et l'ajustement objectif à l'aide des techniques d'apprentissage comme la Recherche dans l'Espace des Poids (REP), la Régression Multi-Linéaire (RML) et les Algorithmes Génétiques (AG). Il existe aussi, les approches de réglage subjectif qui intègrent la préférence humaine dans le processus d'ajustement du poids. Ou bien l'hybridation de ces deux derniers pour bénéficier du jugement humain d'un côté et éviter le problème de la fatigue et de la frustration des utilisateurs d'un autre côté, par les mesures objectives. Au cours de ces dernières décennies, le terme « Coran digital » est adopté dans plusieurs applications qui sont, principalement, la lecture, la sécurité et l'authentification du Coran et l'enseignement des règles de récitation (règles de *tajweed*). Les systèmes dédiés à la récitation du Coran se trouvent sous forme d'appareils commerciaux, comme les stylos de récitation, ou des logiciels Web ou téléphoniques. Généralement, ceux-ci reposent sur le stockage de toutes les *sourates* et versets du Coran, et limitent l'utilisateur d'écouter par mot ou par un verset complet. Par conséquent, notre objectif principal est de surmonter ces limitations en utilisant la synthèse vocale. Le développement d'un système de synthèse pour la récitation du Saint Coran « Holy Quran Text-To-Speech : HQ_TTS » va diminuer l'espace mémoire occupé et donner aux utilisateurs la liberté de choisir la partie qu'ils souhaitent écouter.

Pour atteindre notre objective d'élaboration du système HQ_TTS avec de bonnes performances, notre contribution consiste à :

- la construction d'une BD contient 11077 unités acoustiques dans différents contextes prosodiques et linguistiques ;
- le développement d'un module de transcription automatique pour le Saint Coran, et son optimisation en se basant sur les règles et caractéristiques de ce dernier ;
- la proposition d'une nouvelle approche pour la pondération des caractéristiques utilisées dans la fonction de coût. Ceci par l'utilisation de l'outil des Systèmes Experts (SE) et l'emploi des spécificités phonétiques et phonologiques de la

langue arabe et du SC ;

- la subdivision de l'algorithme de sélection conventionnel en deux parties. Ainsi, qu'une recherche par programmation dynamique forward-backward pour la sélection des meilleures unités.

Cette thèse est organisée comme suit :
Le premier chapitre décrie la parole, la langue arabe et le domaine de la synthèse de la parole. Le deuxième aborde les principes de la SSU ainsi que quelques outils utilisés comme les systèmes experts et les algorithmes génétiques. Dans le troisième chapitre, nous présentons les étapes de développement du système HQ_TTS. Tandis que le quatrième présente les tests et évaluations effectués sur le HQ_TTS et leurs résultats. Enfin nous compléterons notre thèse par des conclusions et quelques recommandations et perspectives pour améliorer ce travail.

CHAPITRE 1 : NOTIONS FONDAMENTALES SUR LA SYNTHÈSE DE LA PAROLE ET LA LANGUE ARABE

1.1 Introduction

La synthèse de parole est un axe de recherche qui interchange plusieurs domaines: l'électronique, l'informatique, la linguistique, la physiologie, l'acoustique et la phonétique. Dans ce premier chapitre, nous allons présenter les principales notions et connaissances nécessaires pour construire un système de synthèse en langue arabe. D'abord, nous commencerons par une généralité sur la parole et certains de ses niveaux de descriptions. Ensuite, nous allons donner une brève description sur l'Arabe et ses principales caractéristiques. Nous finirons par un aperçu sur la synthèse de parole, ses principales techniques et méthodes, ainsi qu'est-ce qu'un système de synthèse à partir du texte et comment peut-on l'évaluer.

1.2 Généralités sur la parole

La parole est un acte essentiel et principal pour la communication entre les humains. Elle résulte et perçoit par des actions volontaires et coordonnées d'un certain nombre d'organes sous le contrôle du système nerveux central.
Le Traitement Automatique de la Parole (TAP) est un domaine de recherche actif en croisement du traitement du signal et du traitement du langage. Le TAP se fait dans plusieurs niveaux de description : physiologique, phonétique, phonologique, acoustique, morphologique, sémantique et pragmatique. Dont l'étude de ces derniers dépend de la tâche ciblée : analyse, reconnaissance, synthèse ou codage. Dans cette thèse, nous nous intéressons aux niveaux de description suivants [1, 2].

1.2.1 Niveau physiologique

Il s'intéresse à l'anatomie des organes responsables de la production de la parole (figure1.1). Cette dernière est basée, généralement, sur trois mécanismes [2]:

- la respiration : dans cette phase, les poumons génèrent l'énergie nécessaire en forme de flux d'air (air pulmonaire) qui se produit lors de l'expiration. Cette phase agit sur l'intensité et la durée du signal vocal ;
- la phonation : puis, le flux d'air provient des poumons à travers la trachée, rencontre le larynx qui sert à modifier et moduler ce flux à l'aide des cordes vocales. La vibration de ces derniers, lors de passage de l'air, produit la principale caractéristique des sons voisés, qui est la fréquence fondamentale du signal (F_0);

- l'articulation : après le passage par les cordes vocales, l'aire pulmonaire rencontre le conduit vocal qui couvre le secteur de pharynx et les cavités buccale et nasale. La forme de ce conduit (définie par la position des articulateurs tels que la langue, la mâchoire, les lèvres, etc.) détermine le timbre des différents sons de langage.

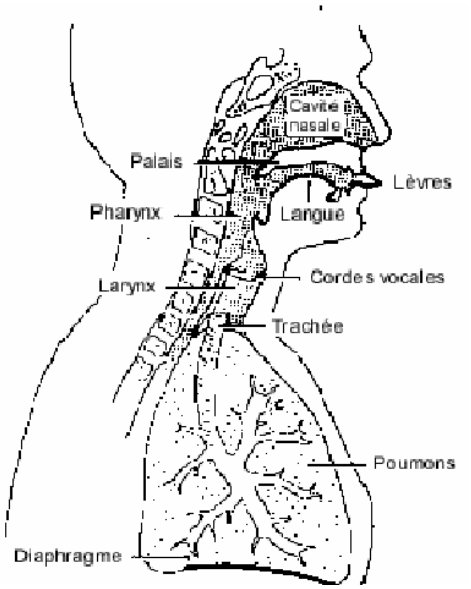

Figure 1.1: Anatomie de l'appareil phonatoire humain [2]

1.2.2 Niveau acoustique

Dans l'acoustique, nous nous intéressons à l'étude du signal de la parole. D'abord, nous le transforme en un signal électrique, puis nous le traite par des méthodes statistiques, pour extraire les principaux traits acoustiques, qui sont [2, 3].

- la fréquence fondamentale : c'est la fréquence de vibration des cordes vocales. Elle se diffère d'une personne à une autre en fonction de la taille des cordes vocales, et elle se divise en trois grandes classes : de 70 à 250 Hz pour les hommes ; de 150 à 400 Hz pour les femmes et de 200 à 600 Hz pour les enfants ;

- la durée : c'est le temps de prononciation d'un phonème. Deux types de la durée peuvent être distingués :

  – la durée observée : qui correspond à la mesure objective du temps de l'activation des organes de phonation ;

  – la durée perçue : qui est liée au mécanisme de la perception. Elle est fréquemment utilisée dans le cas des sons occlusifs puisqu'ils sont caractérisés par une durée de réalisation non continue.

- l'intensité ou l'énergie : qui est liée à la pression d'aire en amont du larynx. Généralement, elle exprime le volume sonore d'un phonème ou l'amplitude des vibrations des cordes vocales dans les sons voisés.

1.2.3 Niveau phonologique

La phonologie est la science qui étudie les sons d'un langage sous l'angle de leur utilisation et leur fonction linguistique. Nous distinguons deux grands domaines de la phonologie [4] :

- la phonématique, qui étudie les unités sonores distinctives minimales (les phonèmes) dans chaque langue, et les traits distinctifs ou traits pertinents qui opposent entre les différents phonèmes d'une même langue ;

- la prosodie, qui est le champ d'étude des phénomènes accompagnants la réalisation de deux ou plusieurs phonèmes. Ces phénomènes s'appellent traits prosodiques ou suprasegmentaux et comprennent [5, 6, 7]:

  – l'accent : consiste en la mise en valeur d'une syllabe, qui se présente par une augmentation de la longueur (la durée du son) et de l'intensité de la syllabe. En Arabe :

    * elle se met sur la dernière syllabe du mot si elle est une surlongue comme : يدرسون [jadrusuun] ;
    * si elle n'est pas sur la dernière syllabe, elle se met sur l'avant-dernière, sauf celle-ci est brève. Exp :نـظّارة [nazzaarah]
    * sinon, c'est sur l'avant-avant-dernière syllabe. Exp :مدرسة [madrasah] ;
    * dans les mots de deux syllabes qui ne correspondent pas au premier cas, l'accent se met à la première syllabe. Exp :ذَهَب [ðahab].

  – le ton : c'est la qualité de la voix en termes de hauteur, de timbre et d'intensité ;

  – l'intonation : c'est la mélodie de la phrase. Elle est, essentiellement, reconnue par la modulation de la voix à l'intérieur de la phrase (la variation de la hauteur de la voix), l'intonation comporte :

    * le rythme : il est créé par l'alternance plus ou moins régulière des syllabes accentuées, des syllabes inaccentuées et des pauses. Lors d'une communication orale, nous pouvons lui donner diverses

caractéristiques : calme et posé, rapide et dynamique, saccadé et nerveux ;

* le débit : c'est la vitesse de prononciation de la phrase ou mot.

1.2.4 Niveau phonétique

La phonétique est l'étude scientifique des sons du langage humain, sous l'angle de leur production, propagation et réception. Nous distinguons trois branches de la phonétique :

1.2.4.1 Phonétique auditive

Elle s'intéresse à l'étude de tout ce qui concerne le processus de l'audition et la façon dont l'être humain perçoit et reconnaît les sons. L'oreille humaine ne répond pas à n'importe quelle fréquence. Elle entend les sons entre 20 et 20000 Hz, avec un maximum de sensibilité dans la plage [250, 8000] Hz. De plus, elle n'entend pas au-dessous 0 dB à 10 m (seuil d'audition), et elle a un seuil de douleur environ 120 dB qu'elle ne supporte aucun son au-dessus de lui (figure 1.2) [2, 6, 8].

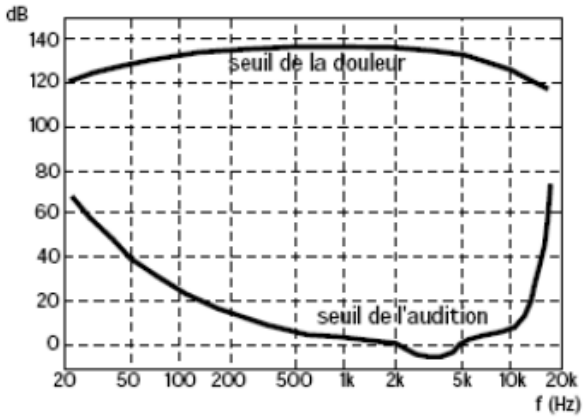

Figure 1.2: Plage d'audition de l'oreille humaine [9]

1.2.4.2 Phonétique articulatoire

C'est l'étude et la classification des phonèmes en fonction de leur mode et lieu d'articulation, où nous distinguons trois grandes classes [2, 8]:

- les voyelles qui sont caractérisées par la vibration des cordes vocales, et que l'aire pulmonaire travers le conduit vocal librement sans obstacle. L'Arabe comprend trois voyelles courtes [a], [u], [i] et trois autres longues [aa], [uu], [ii]. Ces derniers se diffèrent par rapport aux autres, comme leur nom décrit, par une durée plus grande. Comme indique la figure 1.3, l'Arabe est caractérisé par des

voyelles orales qui se distinguent par le degré d'ouverture de conduit vocal, et la position de constriction de la langue ;

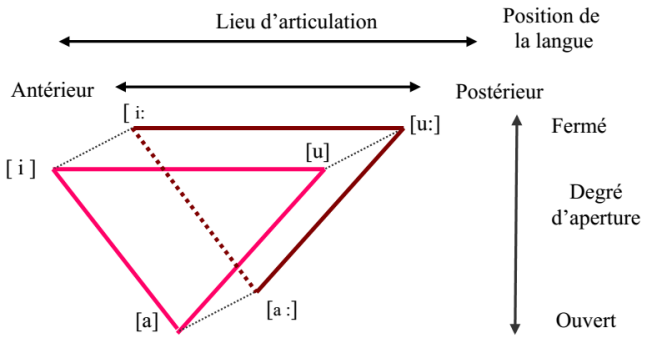

Figure 1.3: Triangle vocalique de la langue arabe [10]

- les consonnes qui sont des sons se produisent soit par une fermeture partielle ou totale du conduit vocal, appelées consonnes occlusives (plosives). Ou bien par une constriction du conduit vocal qui sont les consonnes constrictives (fricatives). Elles se diffèrent entre eux par leurs :
  - mode d'articulation : qui est la façon de génération de la consonne (fricatives ou occlusives) ;
  - lieu d'articulation : principalement, nous distinguons cinq lieux : la gorge, la langue, le palais, le nez (la cavité nasale) et les lèvres (figure 1.4);
  - nasalisation : consonnes nasales ou orales ;
  - voisement : consonnes sonores qui engendrent la vibration des cordes vocales ou consonnes sourdes.

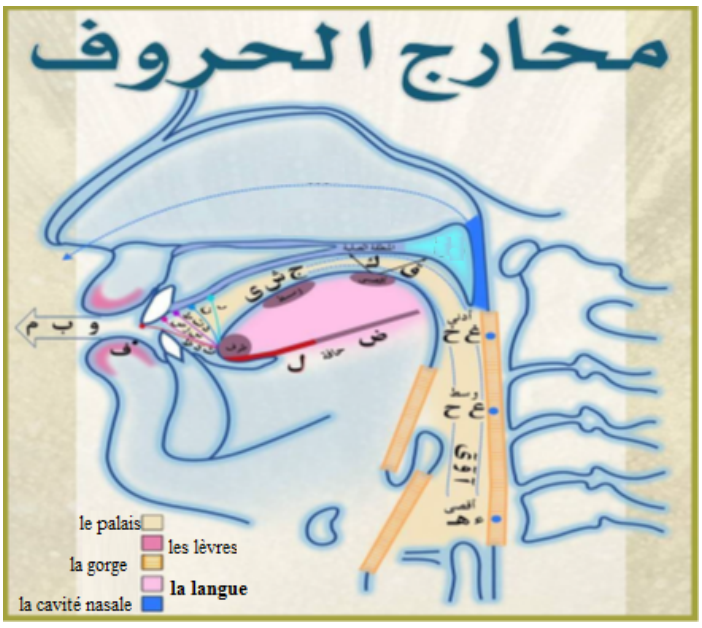

Figure 1.4: Lieux d'articulation des phonèmes arabes [11]

- les semi-voyelles : ils sont des sons qui combinent certaines caractéristiques des voyelles et des consonnes. Leur position centrale est assez ouverte, comme les voyelles, mais le relâchement soudain de cette position produit une friction, comme les consonnes.

La langue arabe comprend 26 consonnes et 2 semivoyelles correspondent, chacune, à un phonème. La figure 1.4 décrit les lieux d'articulation des sons arabes, tandis que le tableau 1.1 présente leurs caractéristiques principales.

Tableau 1.1: Voisement, mode et lieu d'articulation des consonnes et semi-voyelles Arabe [13]

Lettre en AS	**API**	**Mode d'articulation**	**Voisement**	**Lieu d'articulation**
ا ، ء	[ʔ]		voisé	Glotal
ب	[b]	occlusif	voisée	Bilabial
ت	[t]		Non voisées	Alvéodental
ث	[θ]	fricatif		Interdental
ج	[ʤ]	affriquée	voisée	Alvéodental
ح	[ħ]	fricatif	Non voisées	Pharyngal
خ	[x]			Vélaire
د	[d]	occlusif		Alvéodental
ذ	[ð]	fricatif	voisées	Interdental
ر	[r]	vibrante		Apicvoalvéolaire
ز	[z]			Dorsoalvéolaire
س	[s]	fricatif	Non voisé	Dorsoalvéolaire
ش	[ʃ]			Palatal
ص	[ṣ]		voisées	dorsoalvéolaire
ض	[ḍ]	occlusif		Alvéolaire
ط	[ṭ]		Non voisée	Alvéodentale
ظ	[ẓ]			Interdental
ع	[ʕ]	fricatif	voisées	Pharyngal
غ	[ɣ]			Uvulaire
ف	[f]		Non voisées	Labiodental
ق	[q]	occlusif		Uvulaire
ك	[k]			Postpalatal
ل	[l]	liquide	voisées	Dental
م	[m]	nasales		Bilabial
ن	[n]			Alvéodentale
هـ	[h]	fricatif	Non voisée	Glotal
و	[w]	Semi-voyelles	voisées	Bilabial
ي	[y]			Palatal

1.2.4.3 Phonétique acoustique

Elle étudie et examine les caractéristiques acoustiques des sons, la fréquence fondamentale, l'énergie, la durée, etc. [8, 12]. Par exemple :

- les voyelles se caractérisent principalement par la présence des maximas spectraux (zones de fréquence intenses) qui s'appellent formants. L'augmentation du premier formant F1 se traduit par l'ouverture de l'articulation, et l'augmentation de F2 par l'antériorisation de l'articulation (figure 1.3);
- les consonnes fricatives sont des évènements apériodiques qui se présentent comme un bruit résulte d'une turbulence aérodynamique qui prend naissance au point d'articulation. Dans un Spectrogramme, avec un filtre de 300 Hz le bruit de turbulence apparaît comme un ensemble de petites tries verticales plus ou moins longues déposées aléatoirement et d'intensité variable (figure 1.5);
- les consonnes occlusives se caractérisent par un silence suivi par une barre d'explosion et à la fin par un bruit de friction. La durée de ces phénomènes acoustique se différencie d'un son à un autre, tout dépend du lieu d'articulation (figure 1.5).

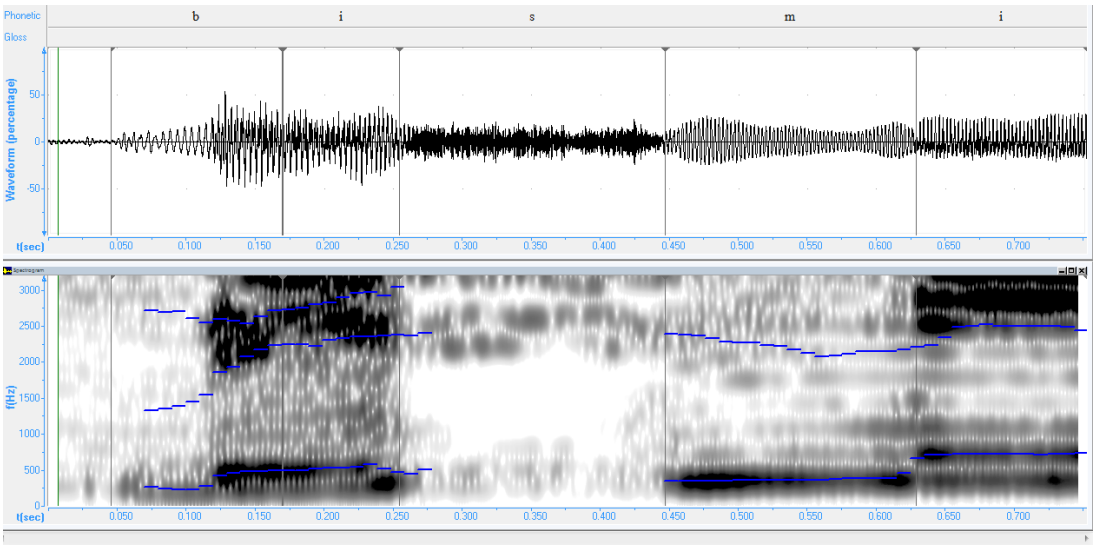

Figure 1.5: Audiogramme et spectrogramme du mot [bismi] qui présente quelques caractéristiques acoustiques des sons

1.3 Langue arabe

L'Arabe est la langue la plus parlée des langues sémitiques comme l'Araméen, l'Amharique, l'Hébreu, etc. C'est la langue principale de 23 pays du Moyen-Orient

et l'Afrique du Nord. Elle est parlée par plus de 422 millions de personnes, et plus de 1,62 milliards de musulmans dans le monde l'utilisent. Dans la vie courante, nous utilisons deux types de langages arabes : le dialectal qui se diffère d'une région à une autre, et l'Arabe Standard (AS) dont nous nous travaillons. L'AS est la langue du Saint Coran (SC), enseignée dans les écoles, écrite dans la littérature et utilisée dans le cadre officiel. C'est celle qui réunit la parole de tous les pays arabes. Comme toute autre langue, l'Arabe comprend des caractéristiques propres à elle comme.

1.3.1 Signes diacritiques

Ils sont des caractères spécifiques qui s'ajoutent aux lettres de l'alphabet arabes pour enlever l'ambiguïté de la lecture et pour mieux comprendre le contexte, les principaux signes diacritiques sont (figure 1.6) :

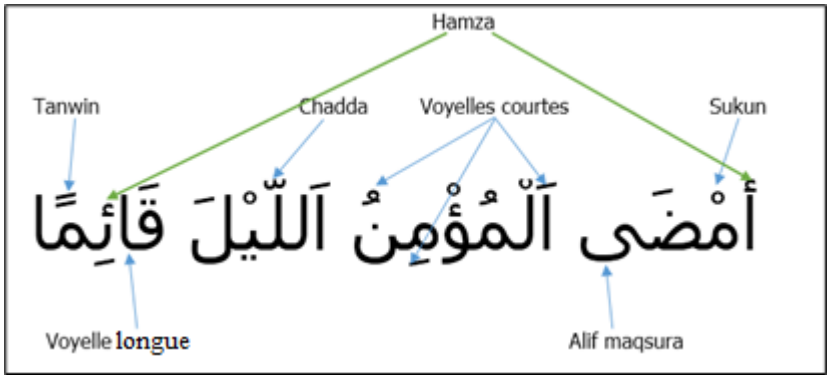

Figure 1.6: Exemple des signes diacritiques de l'AS

- les *harakaat* : la *fatha* «ـَ» , la *dhamma* «ـُ» et la *kasra* «ـِ» , elles sont placées au-dessus ou au-dessous les consonnes pour représenter respectivement les voyelles courtes [a] , [u] et [i]. Ces signes précèdent aussi les consonnes و ، ا et ي pour avoir respectivement les voyelles longues [uu], [aa] et [ii] ;

- le *sukune* «ـْ», il s'écrit au-dessus de la consonne lorsqu'elle n'est pas munie d'une voyelle. Il n'affecte pas une consonne au début du mot et il ne suit pas une voyelle longue ;

- la nounation (*tanween*) : c'est un dédoublement d'une voyelle courte à la fin du mot. Phonétiquement, elle se prononce comme un ajout du phonème [n] à la voyelle prononcée. Les différentes formes de *tanween* référant au mot قَمَرْ [qamar] : «ـًا» [an] : قَمَرًا [qamaran] ; «ـٌ» [un] : قَمَرٌ [qamarun] ; «ـٍ» [in] : قَمَرٍ [qamarin] ;

- la *chadda* «ـّ» : elle se place au-dessus d'une consonne non initiale et signifie son dédoublement. إِنَّ [ʔinna] ;

- le *madd* qui est une transcription du signe (~) au-dessus du caractère (ا) remplaçant la voyelle longue [aa]. آمَنَ [ʔaamana] ;

- la *hamza* : certain chercheurs considèrent le caractère (ء) comme la 29 ème consonne de l'AS. Ceci parce qu'il ne se présente pas, souvent, elle avec le *alif* qui est la première consonne de l'AS. Tandis que d'autres chercheurs considèrent la *hamza* comme un signe diacritique qui s'accompagne avec la consonne *alif* [ʔ]. Dans ce dernier cas, son apparition et ses modes d'écriture se basent sur des règles grammaticales précises.

1.3.1.1 Caractéristiques phonologiques

La langue Arabe présente des caractéristiques phonologiques que nous devons les tenir en compte dans n'importe quel système de TAP Arabe. Certaines caractéristiques sont spécifiques à la récitation de Saint Coran, et d'autres nous les trouvons dans l'AS ou dans les dialectes arabes comme [14, 15] :

- l'emphase : c'est un phénomène co-articulatoire spécifique aux langues sémitiques et plus particulièrement l'Arabe. Plusieurs manifestations entrent dans cette appellation comme : la pharyngalisation, l'uvularisation et la glottalisation [15]. Chez les linguistes arabes la pharyngalisation désigne les consonnes ayant les caractéristiques de : l'*al-itbâq*, *al-isti'alâ* et le *tafkhim* (traduit par une expression acoustique grasse et épaisse de certaines consonnes). Ces consonnes sont : [s̩], [d̩], [t̩], [z̩]. Puisque l'emphase est un phénomène coarticulatoire, il influence le timbre des voyelles adjacentes (courtes ou longues). Ces nouvelles variantes se caractérisent par le rapprochement des formants F_1 et F_2 à cause de l'abaissement de F_2 (corrélat acoustique de la postériorisation). Certaines études ont monté que l'influence de l'emphase peut dépasser la voyelle (ou les voyelles) adjacente(s) [14] ;

- la gémination : c'est un phénomène signifie le dédoublement de deux consonnes identiques en une seule dite géminée. Toutes les consonnes arabes acceptent la gémination sauf la *hamza*. Sur le plan phonétique, ce phénomène agit beaucoup sur la durée de la consonne géminée (augmentation ou presque doublement de la durée totale de la consonne no occlusive et de la durée de silence pour les occlusives). Ce rallongement entraîne l'accentuation des propriétés de la consonne, comme l'augmentation du caractère emphatique. Contrairement aux

autres langues, la gémination est un élément distinctif qui joue un rôle très important dans le plan morphosémantique de la langue arabe. L'exemple : دَرَسَ : [darasa] "il a étudié" est différent de دَرَّسَ [darrasa] "il a enseigné" où la deuxième consonne est géminée.

1.4 Synthèse de la parole

La synthèse de la parole est la génération (production) artificielle de la parole par une machine. Selon le type des données entrées, nous distinguons deux classes de synthétiseurs. Celle qui reproduit la parole à partir des caractéristiques numériques d'un signal vocal et les synthétiseurs de parole à partir d'une représentation symbolique [16]. Le premier essai d'un synthétiseur vocal était en 1780 par Kratzenstein avec un système basé sur le principe des boîtes de musique qui est capable de prononcer cinq voyelles. Cependant, la véritable initiative d'un système de production de la parole revient à Von Kemprlen en 1791. Ensuite, les essais ont été continués par l'amélioration de ce dernier système comme celle de Joseph Faber entre 1830-1840 (*Euphonia*), de Charles Wheatstone en 1835 (figure 1.7), etc.

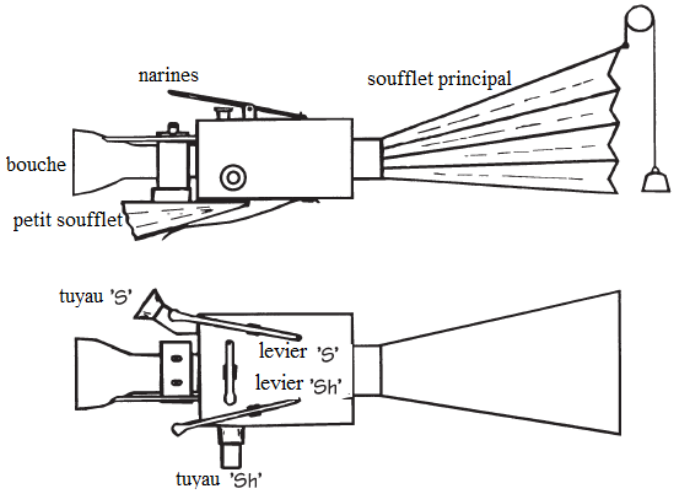

Figure 1.7: Reconstruction de la machine de Kempelen par Wheatstone [8]

Au XX^{ème} siècle, l'apparition de l'électricité et de l'électronique a donné une autre vision sur la synthèse de la parole et a motivé les chercheurs d'améliorer et construire des nouveaux systèmes comme : la machine de J.Cl.Stewart (1922) ; le VODER (Voice Operation Demonstrator) de Homer Dudley de laboratoire Bell en 1939 (figure 1.8) ; l'utilisation des rayons X dans la production et la transmission de la parole ; et la construction des premiers analogues électrique du conduit vocale (Dunn, 1950 ; Steven et al 1953) [2, 12]. À partir des années 70, la synthèse vocale a été passée à une étape plus élevée avec le début de numérisation et l'utilisation des calculateurs.

Pour cela, des nouvelles méthodes et techniques ont vu le jour et ont connu un grand développement. Ces nouvelles technologies ont rendu la parole synthétique très intelligible et elle devenu de plus en plus naturelle, expressive et variable (facile à changer le sexe, la voix, etc.).

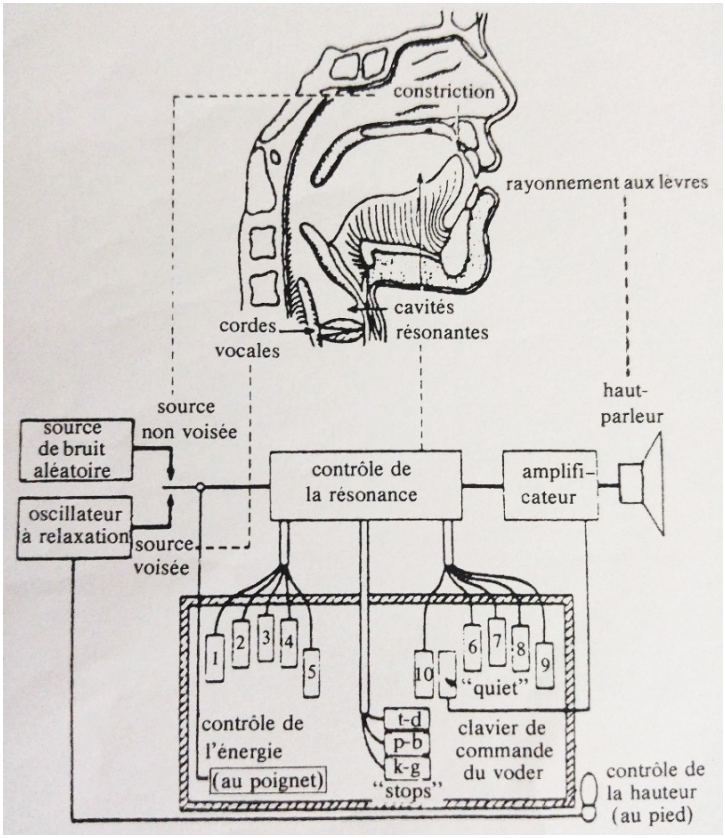

Figure 1.8: Diagramme de VODER [12]

1.4.1 Techniques d'analyse de la parole

La parole est un signal réel, d'énergie finie et non stationnaire. Ce signal peut être pseudopériodique pour les sons voisés, aléatoire pour les fricatifs ou impulsionnel pour les sons occlusifs. Avant toute opération de synthèse une étape d'analyse du signal est nécessaire pour extraire et étudier ses paramètres pertinents. Les techniques d'analyse du signal vocal peuvent se classer suivant leur domaine d'utilisation, temporelle ou spectrale [2, 10, 12, 17, 18]. Elles peuvent s'appliquer étape par étape selon leur description, ou bien à l'aide des outils (logiciels) spécial comme : Praat, speech analyzer, SFS, etc.

1.4.1.1 Techniques temporelles

Ils sont des formules qui s'appliquent à la version temporelle du signal vocal comme:

1.4.1.1.1 Transformé de Fourier (TF)

Dans le domaine de TAP, nous utilisons la TF dite à court terme, à cause de la non stationnarité du signal vocal. Cette TF est obtenue par l'extraction d'une trentaine de milli seconds du signal, ensuite l'application d'une fenêtre de pondération (souvent une fenêtre de Hamming). À la fin, la TF est effectuée sur les échantillons résultants, comme présente l'équation (1.1). La TF est utilisée pour obtenir le spectre du signal vocal, où les parties voisées du signal apparaissent sous forme d'une succession de pics spectraux marqués, dont les fréquences centrales sont multiples de la fréquence fondamentale. Par contre, les parties non voisées ne présentent aucune structure particulière.

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-\frac{2\pi i}{N} kn} \tag{1.1}$$

tel que :

$k = 0, .., N-1$;

$x[n]$une trame du signal numérique de longueur N;

$X[k]$la Transformé en Fourier Discrète (TFD) de $x[n]$.

1.4.1.1.2 Fonction d'autocorrélation

Elle est utilisée pour déterminer le pitch du signal vocal par la détection des maximas de cette fonction. La position de ces maximas nous informe sur l'existence du fondamental d'un signal. La fonction d'autocorrélation, comme calculée dans l'équation (1.2), ne s'applique qu'après un filtrage de préaccentuation pour accentuer la partie haute fréquence. Comme la TF, elle nécessite aussi une division en trames et une fenêtre de pondération à cause de la non stationnarité du signal.

$$r[k] = \sum_{n=1}^{n-1+k} x[n].x[n+k] \tag{1.2}$$

1.4.1.1.3 Taux de passage par zéro « Zero-Crossing Rate (ZCR) »

C'est le taux de changement de signe d'un signal (le passage de la partie positive à la négative ou inversement). Il est utilisé pour déterminer la partie voisée, caractérisée par des petites valeurs de ZCR, de la quel non voisée, suivant la formule :

$$ZCR = \frac{1}{T-1} \sum_{t-1}^{T-1} II(S_{t-1} < 0) \tag{1.3}$$

où : S est un signal de longueur T et $II(A)$ est une fonction indicative prend la valeur "1" si A est vrais et "0" autrement.

1.4.1.1.4 Energie

Les informations extraites à partir de l'énergie du signal vocal peuvent être limitées, mais utile. Par exemple l'amplitude de signal dans les zones non voisées est notamment inférieure au celles des zones voisées.

1.4.1.2 Techniques spectrales

Ils sont des traitements qui se font à la version spectrale du signal vocal comme :

1.4.1.2.1 Spectrogramme

C'est une représentation de l'évolution temporelle du spectre de parole, dont l'énergie du signal apparaît sous forme de niveau de gris dans un diagramme en deux dimensions temps-fréquence. Pour avoir cette représentation, nous devons suivre les étapes suivantes :

- diviser le signal en une suite de trames de petites durées (environ 10 ms) ;
- appliquer la Transformé de Fourier Rapide « Fast Fourier Transform (FFT) » sur chacune de ces trames ;
- positionner chaque spectre sur le centre de chaque trame ;
- coder l'amplitude de spectre en niveaux de gris.

Dans cette représentation, nous distinguons les spectrogrammes à large bande et à bande étroite tout dépend de la durée de fenêtre de pondération. Les spectrogrammes à large bande (figure 1.9 a) sont obtenus avec des fenêtres de pondération de faible durée (typiquement 10 ms). Ils permettent de visualiser l'évolution temporelle des formants. Tandis que, les spectrogrammes à bande étroite (figure 1.9 b) sont moins utilisés, et ils mettent en évidence la structure fine du spectre.

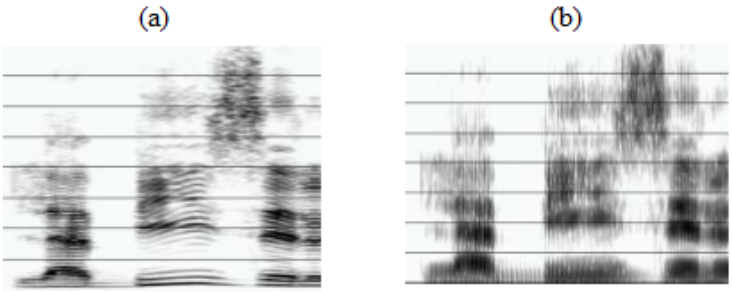

Figure 1.9: Spectrogrammes (a) à large bande et (b) à bande étroite [18]

1.4.1.2.2 Codage prédictif linéaire « Linair Predictive Coding (LPC) »

La LPC est une méthode d'analyse largement utilisée comme un modèle estimateur des paramètres de la parole dans le codage, la synthèse et la reconnaissance de celle-ci. Son principe réside que chaque échantillon de parole s[n] peut se prédire par une combinaison linéaire d'un nombre fini des p instants précédents ($s[n-l]$ à $s[n-k]$) avec un petit terme d'erreur $e[n]$ appelé signal résiduel, comme suit.

$$s[n] = \sum_{k=1}^{p} a_k s[n-k] + e[n] \tag{1.4}$$

où

p : l'ordre de la prédiction linéaire ;

a_k les coefficients de la prédiction linéaire.

Dans la Reconnaissance Automatique de la Parole (RAP) les coefficients ak sont utilisés comme des paramètres pour la modélisation du signal vocal. Tandis que dans les synthétiseurs, les ak sont considérés comme des valeurs à prédire. Ces valeurs sont calculées par la minimisation de l'erreur quadratique moyenne entre le signal original $s[n]$ et le signal prédit $\tilde{s}[n]$sur une fenêtre donnée (équation 1.5). Les coefficients sont réactualisés régulièrement toutes les 5 à 20 ms.

$$\sum_{n} e^2[n] = \sum_{n=0} (s[n] - \tilde{s}[n])^2 = \sum_{n=0} (s[n] - \sum_{k=1}^{p} a_k s[n-k])^2 \tag{1.5}$$

1.4.1.2.3 Cepstre

C'est une transformation homomorphique du signal vocal, du domaine temporel vers un autre domaine analogue. Son utilité réside dans la séparation de la contribution du conduit vocal de l'onde glottique (c.à.d. séparer la fréquence fondamentale des modifications qu'elle subit dans le conduit vocal « les harmoniques »). La parole est le résultat d'une convolution du signal de la source par le filtre correspondant au conduit vocal (équation 1.6). Par conséquent, la séparation des deux contributions se fait par déconvolution. Par application de la TF, la convolution se transforme en produit, et le logarithme transforme ce produit en une somme, comme présente les équations (1.7) et (1.8) :

$$s(n) = u(t) * h(t) \tag{1.6}$$

avec :

$s(t)$: le signal temporel ;

$u(t)$: le signal excitateur (de la source) ;

$h(t)$: la contribution du conduit.

$$S(f) = U(f) \times H(f) \tag{1.7}$$

$$\log(|S(f)|) = \log(|U(f)|) + \log(|H(f)|) \tag{1.8}$$

Par une transformation inverse de ce dernier signal, nous obtenons le cepstre, et nous aurons une relation dans le domaine temporelle donnée par :

$$TF^{-1}(\log(|S(f)|)) = TF^{-1}(\log(|U(f)|)) + TF^{-1}(\log(|H(f)|)) \tag{1.9}$$

Donc pour une trame d'un signal de parole $x[n]$ les coefficients cepstraux $c[n]$ sont donnés par :

$$c[n] = \sum_{k=0}^{K-1} \log(|\sum_{n=0}^{N-1} x[n] e^{-\frac{2\pi i}{N} kn}|) e^{\frac{2\pi i}{K} kn} \tag{1.10}$$

La variable indépendante du cepstre est, nominalement, le temps puisqu'il est la TFD Inverse (TFDI) d'un spectre. Cependant, il s'interprète aussi comme une fréquence, car le logarithme du spectre se considère comme une onde. Pour clarifier cette interchangeabilité entre les deux domaines, Bogert, Healy et Tukey en 1960 ont formulé le terme "cepstre" par inversement de l'ordre des premières lettres du mot spectre et par analogie :

$$\begin{cases} Frquence => qufrence; \\ harmonique => rahmonique; \\ magnitude(amplitude) => gamnitude; \\ phase => saphe; \\ filtre => liftre. \end{cases}$$

L'analyse de cepstre permet de séparer la contribution de la source (comprenant la fréquence fondamentale), de l'enveloppe spectrale (la contribution du conduit vocal). Cette dernière, peut se retrouver par l'application d'un liftre passe-bas. Tandis que, le liftrage passe-haut nous renseigne sur le pitch. Si le signal d'entrée possède une période de hauteur fondamentale forte, elle apparaît dans le cepstre sous forme de pic, et par la mesure de la distance entre le temps zéro et le temps pic nous trouvons la période fondamentale de cette hauteur.

Coefficients MFCC (Mel Frequency Cepstral Coefficients)

Les MFCC sont des coefficients cepstraux représentés dans une échelle dite mel.

La représentation dans ce dernier est inspirée du système auditif humain qui a une sensibilité logarithmique diminue quand la fréquence augmente. Le calcul des valeurs MFCC se fait en suivant les étapes ci-dessus (figure 1.10) :

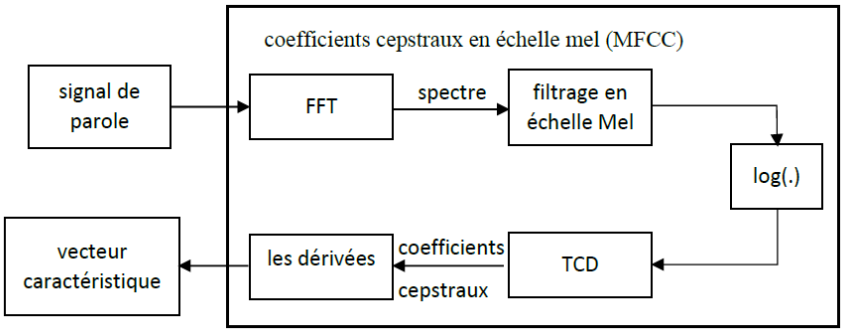

Figure 1.10: Diagramme général de calcul du vecteur caractéristique des valeurs MFCC et leurs dirivées

- la préaccentuation : afin de mieux analyser et modéliser le signal de la parole un filtre de préaccentuation doit être s'appliqué pour l'énergie des hautes fréquences. Car dans la plupart des sons voisés l'énergie tend à chuter dans les hautes fréquences à cause de la nature de l'impulsion glottique (figure 1.11). Le filtre utilisé est de type passe-haut de premier ordre,

$$H(z) = 1 - \alpha z^{-1} \tag{1.11}$$

  avec : $0.9 \leq \alpha \leq 1.0$;

- la division en trames : à cause de la non stationnarité du signal vocal, il faut le segmenter en morceaux d'environ 20 - 30 ms avec un chevauchement de 50% des fenêtres de segmentation ;

- l'application de la TF : calculer le spectre de chaque trame à l'aide d'une TFD: $x[n] -> X[F]$;

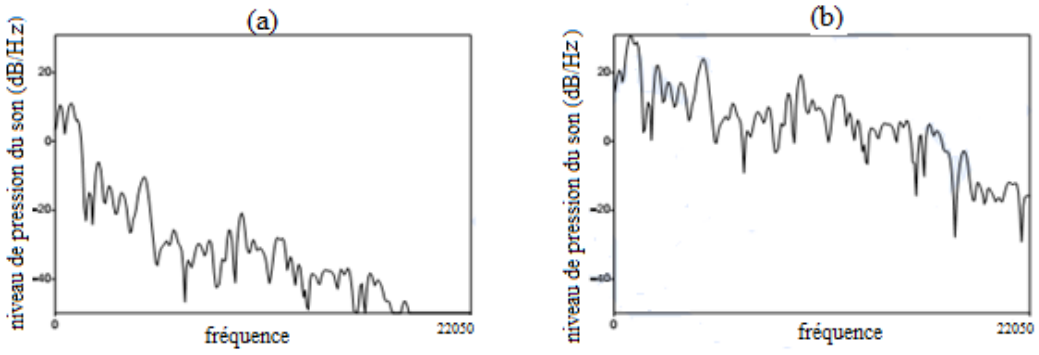

Figure 1.11: Spectre de la voyelle « aa » : (a) avant préaccentuation ; (b) après accentuation [19]

- le changement d'échelle : la TFD nous résulte l'énergie de signal dans le domaine fréquentiel, mais à cause de la sensibilité du système auditive humain par rapport aux fréquences, cette présentation doit être faite dans l'échelle mel ;

$$mel(f) = 1127 \ln(1 + \frac{f}{700}) \tag{1.12}$$

Ce changement d'échelle se réalise par la multiplication du spectre par un banc de filtres tout en collectant l'énergie à chaque band de fréquences (équation 1.13). Les filtres de ce banc sont distribués dans l'échelle mel comme présente la figure 1.12. Ils sont caractérisés par un largueur de band progressive, plus de filtres dans les basses fréquences et moins de filtres dans les hautes fréquences ;

$$E[k] = \sum_{f} W_k[f]|X[f]|^2 \tag{1.13}$$

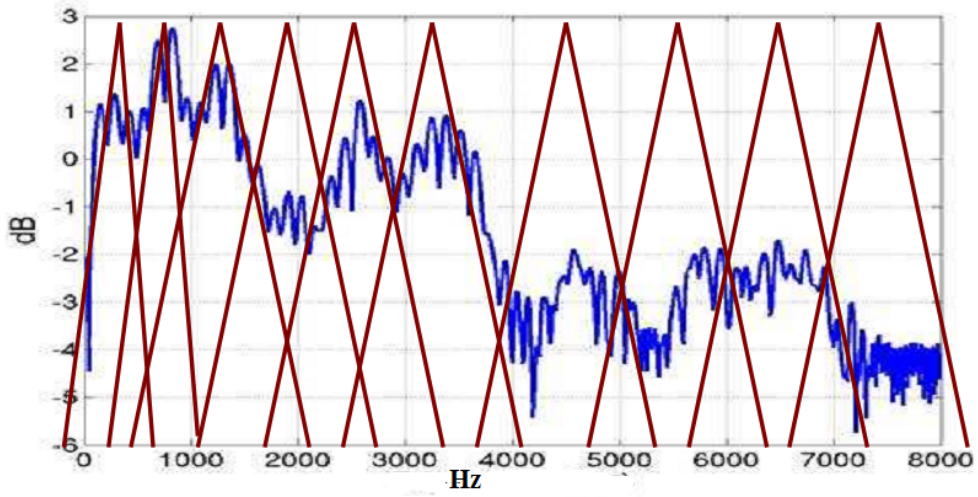

Figure 1.12: Banc de filtres utilisé dans le calcul des MFCC [20]

- le calculer du logarithme de chaque valeur mel spectrale ;

- le passage au cepstre et de trouver les coefficients cepstraux par une Transformé en Cosinus Discret (TCD). La TCD est utilisée au lieu de la TFDI, car ce calcul se fait qu'avec les coefficients de valeurs réelles, qui est plus facile avec la TCD. De plus, la TCD permit la décorrélation entre les coefficients, à cause du chevauchement des bancs de filtres ;

$$c_n = \sum_{k=1}^{K} \log(E[k]) \cos[n(k - \frac{1}{2})\frac{\pi}{M}] \tag{1.14}$$

tel que : $n = 1, .., K$.

Généralement les 12 premiers coefficients sont suffisants pour représenter une

trame. Dans certains cas et pour plus d'informations dynamiques les dérivées première et seconde de ces coefficients sont aussi utilisées (les coefficients delta et delta delta).

1.4.2 Techniques de synthèse de la parole

La synthèse de la parole se base sur des techniques bien adoptées, qui se divisent par rapport au [2, 12, 17]:

- stratégie adoptée : système-modèle qui est des techniques basées sur la modélisation du système phonatoire , ou bien signale-modèle qui est basé sur la modélisation du signal vocal ;
- domaine de traitement du signal vocal : spectral, temporel ou bien articulatoire.

Parmi les techniques existantes nous citons :

1.4.2.1 Codage prédictif linéaire

Cette technique se base sur la modélisation LPC du signal de la parole. Pour générer une parole artificielle, l'algorithme de synthèse a besoin d'un signal d'excitation (un train d'impulsion pour les sons voisés, ou un bruit blanc pour les non voisés) et les coefficients de filtre de prédiction. Le signal d'erreur résultant durant chaque étape peut s'introduire pour rapprocher au signal original. De 10 à 15 coefficients sont nécessaires pour atteindre une synthèse de qualité acceptable (un taux d'échantillonnage de 8 kHz). Et de 20 à 24 coefficients pour avoir une qualité plus haute (un taux d'échantillonnage de 22 kHz). Les inconvénients de cette technique sont, précisément, liés à la source d'excitation et sa représentation qui est trop simple par rapport à la réalité. De plus, le modèle LPC ne représente pas bien les sons nasals et les fricatifs voisés. Un problème peut se poser aussi pour les occlusifs courts quand leur durée est inférieure à la largeur de la trame utilisée pour l'analyse. Pour résoudre le problème de l'excitation, autres extensions de cette technique ont été développées comme :

- la prédiction linéaire par impulsions multiples : utilise une excitation composée de plusieurs impulsions pour chaque trame de parole analysée ;
- RELP Residual Excited Linear Prediction : utilise le signal d'erreur ou résiduel comme signal d'excitation ;

- CELP Codebook Excited Linear Prediction : utilise un dictionnaire de signaux d'excitation .

1.4.2.2 Synthèse par formants

Cette technique est basée sur la génération de la voix artificielle à partir d'un spectrogramme, tout en utilisant les formants du signal vocal. Un synthétiseur à formants est composé de :

- deux sources d'excitations, l'une est un générateur d'impulsions pour les sons voisés et l'autre un générateur de bruit pour les sons non voisés ;
- des blocs de filtres commandés par les fréquences centrales des formants, leurs amplitudes et leurs largeurs de bande. D'autres paramètres et blocs sont aussi ajoutés comme la largeur de bande de bruit pour les non voisés et l'effet du canal nasal. Ces blocs sont associés en série, en parallèle ou de façon mixte.

Un synthétiseur à formant de bonne qualité nécessite de 3 à 4 formants. La corrélation qui existe entre les paramètres utilisés avec le principe de production de la parole, mettre cette technique avantageuse. Cependant, son problème réside dans les techniques d'analyse des formants qui ne sont pas encore satisfaisantes et la plupart des paramètres sont optimisés manuellement.

1.4.2.3 Synthèse articulatoire

Contrairement aux autres techniques qui s'intéressent à l'étude et le traitement de signal dans le domaine temporel ou fréquentiel, cette technique s'appuie sur la simulation du phénomène de production de parole par une modélisation géométrique du conduit vocal. Dans cette technique, nous nous basons sur des principes de la mécanique des fluides, la physique, et la mécanique. La modélisation consiste à représenter le conduit vocal comme un tube de section variable, avec des embranchements et des sections parallèles tout dépend de la position des organes phonatoire (langue, mâchoire, lèvre). Puis, à simuler le trajet des ondes produites au niveau de la glotte. Ce synthétiseur a besoin des paramètres, dits supra-laryngés, pour commander le modèle articulatoire, et autres pour le pilotage des cordes vocales (pression sub-glottique, longueur des cordes vocales et hauteur de la glotte au repos). Cette technique apparaît séduisante par rapport à la qualité de synthèse, car elle rapproche à la réalité physique de production de parole, mais elle est difficile à mettre en œuvre et très coûteuse.

1.4.2.4 Synthèse par l'algorithme PSOLA (Pitch Synchronous OverLap and Add)

L'algorithme PSOLA est un outil de lissage utilisé après certaines méthodes de synthèse. Ce lissage s'effectue par une modification de la durée et du pitch du segment sonore. D'abord, le signal doit subdiviser en une suite de trames élémentaires, à l'aide d'une fenêtre synchronisée avec le pitch. La largeur de cette fenêtre est choisie de tel sorte que le recouvrement mutuel entre deux signaux élémentaires successifs soit important. Après, la modification de la durée se fait par un ajout ou un retranchement des trames des signaux déjà existants. Tandis que la modification de la fréquence se fait par une augmentation ou réduction de la partie de recouvrement. Les variantes de cette technique se diffèrent entre eux par leur domaine d'application : temporel (Time Domain (TD)-PSOLA) ou fréquentiel (Frequency Domain (FD)-PSOLA) ; ou par le type de signal à modifier : le signal d'erreur (Linear Prediction (LP)-PSOLA), ou des segments synthétisés (Multi-Band Excited (MBR)-PSOLA).

1.4.3 Méthodes de synthèse de la parole

Dans la synthèse de la parole, Il existe des différents moyens utilisés pour passer de la représentation symbolique du texte au signal acoustique comme [1, 14, 16, 21]:

1.4.3.1 Synthèse paramétrique

Appelée aussi synthèse par règle, elle base, principalement, sur la modélisation paramétrique du phénomène de production de la parole. Dans cette technique, le signal vocal doit être d'abord analysé pour extraire et modéliser des paramètres simplifiant sa représentation en petites unités de parole. Le type de ces paramètres est déterminé par la technique utilisée articulatoire, par formants ou statistique. Ensuite, une base de règles contextuelles doit être construite pour décrire la transition entre ces paramètres. Par conséquent, le signal synthétisé est généré à l'aide d'un Vocodeur commandé par ces paramètres et règles.
La synthèse paramétrique statistique est la technique la plus utilisée ces jours dans la synthèse par règles. Comme présente la figure 1.13, cette méthode comporte deux phases : l'entraînement et la synthèse. Durant la première, des modèles statistiques vont être estimés à l'aide des caractéristiques acoustiques de la parole, extraites d'une base d'apprentissage. Chacun de ces modèles est après indexé par des caractéristiques linguistiques (associé par une ou plusieurs règles linguistiques) extraites du texte

d'apprentissage correspondant. Dans la deuxième phase, le synthétiseur se base sur les paramètres linguistiques extraits du texte entré pour chercher le modèle le plus convenable à chaque phrase du texte et déduire ses paramètres acoustiques et prosodiques. Ensuite, ces derniers vont être l'entrée d'un vocodeur pour générer la parole. L'intérêt sur cette technique a commencé après le grand succès des HMM (Hidden Markov Model) dans le domaine de la RAP, et maintenant nous trouvons plusieurs modèles statistiques utilisés comme les réseaux de neurones artificiels et toutes ces variantes.

Une telle approche ne permet pas de restituer un signal de parole, assez, naturel. Son problème réside d'une part de la difficulté à modéliser suffisamment et finement les trajectoires acoustiques ou de trouver le meilleur modèle statistique. Et d'autre part de la modélisation trop grossière du signal glottique. Parmi les avantages de cette méthode, nous citons la grande souplesse d'utilisation et la facilité d'extension. De plus, nous pouvons changer les caractéristiques de la parole synthétisée (l'expressivité, l'identité du locuteur, etc.).

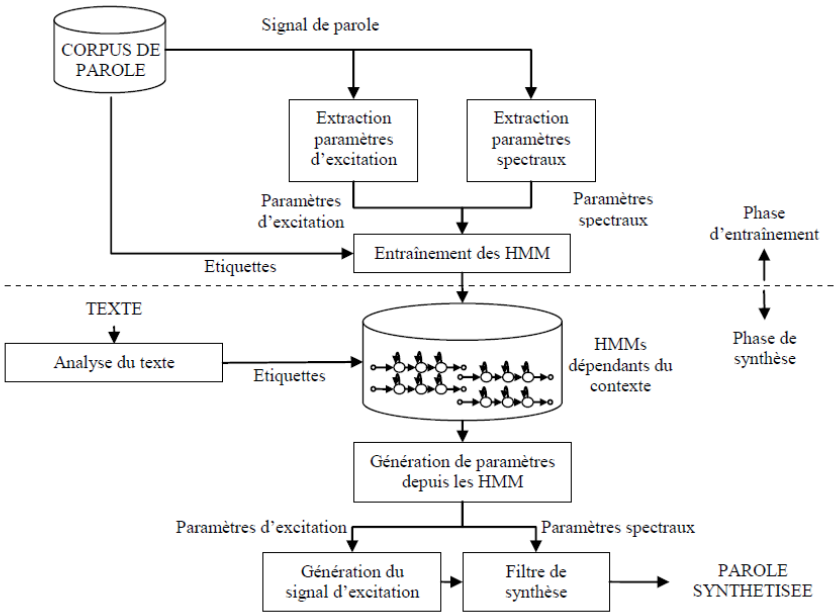

Figure 1.13: Schéma synoptique typique d'un synthétiseur statistique par HMM [21]

1.4.3.2 Synthèse par concaténation

Cette méthode est basée sur la juxtaposition des unités acoustiques pour avoir le signal synthétisé. Ces unités sont des segments sonores naturels extraits d'un corpus de parole préalablement enregistré, composant une Base de Données (BD). Lors de la synthèse les unités à concaténer sont choisies de la BD tout dépend la phrase à synthétiser. Puisqu'elle utilise des segments sonores réels, cette méthode résulte une

parole plus naturelle que la synthèse par règles. De plus, elle nécessite moins de traitement du signal vocal. Dans cette méthode, plusieurs techniques et types d'unité peuvent être utilisés, où nous citons :

- *la concaténation des phrases et mots :* les premiers synthétiseurs par concaténation utilisent des phrases, des mots ou tous les deux combinées. Ces systèmes résultent la parole la plus naturelle, mais ils peuvent s'intégrer que dans des applications à vocabulaire limité à cause de la taille d'enregistrement qu'ils nécessitent comme : les horloges parlantes, les répondeurs téléphoniques, les systèmes d'annonces dans le transport ou dans les files d'attentes. Généralement, la phrase résultante est composée de deux parties, une phrase fixe qui se répète à chaque fois et une autre variable qui change à chaque annonce ;

- *la concaténation des phonèmes :* pour des applications plus générales comme la lecture d'un texte nous avons besoin de synthétiser n'importe quelle phrase ou mot sans qu'ils soient enregistrés au préalable. Celui-là, est réalisable qu'avec des unités plus petites comme le phonème Cependant, la concaténation des phonèmes ne donne pas un bon résultat à cause du phénomène de coarticulation entre les sons qui donne une discontinuité au niveau du son synthétisé. Car, c'est la transition entre les phonèmes qui portent les principales informations de la parole ;

- *la concaténation des diphones :* pour résoudre le problème posé par la concaténation des phonèmes les chercheurs ont pensé à des unités minimales qui couvrent les parties instables dans le signal vocal comme le diphone. Un diphone est un élément sonore comprend la transition entre deux phonèmes. Il s'étendant de la partie stable d'un phonème à la partie stable du phonème suivant ;

- *la concaténation des syllabes et polyphones :* l'utilisation des diphones cause aussi le problème de discontinuité dans le cas des transitions complexes comme les groupes consonantiques des semi-voyelles et liquides. Pour cela, des unités de taille plus grande sont proposées comme la syllabe, la disyllabe des unités contient trois phonèmes ou plus, appelées polyphones. Ces unités sont préférables, car avec l'augmentation de la taille de l'unité on préserve plus les

caractéristiques prosodiques du signal, mais il ne faut pas oublier que cela nécessite une mémoire de stockage plus grande;

- *la Synthèse par Sélection d'Unités (SSU) :* dans le but de plus améliorer le naturel de la parole synthétique, l'idée de cette méthode est basée sur l'enrichissement de la BD par plus des unités de différentes tailles et dans des contextes linguistiques et prosodiques distincts. Ensuite, dans la phase de synthèse, les unités qui vont être concaténées sont celles qui produisent la moindre de discontinuité et leur prosodie est la plus proche de celle souhaitée. Cette technique est capable de synthétiser une parole dont l'intelligibilité et le naturel rendent possible la confusion avec une prononciation humaine, mais elle est gourmande au niveau de la taille mémoire, et elle nécessite un accès très rapide à cette dernière.

1.4.4 Systèmes de synthèse à partir du texte (Text-To-Speech : TTS)

Selon le type d'entrée et le mode d'opération, les systèmes de synthèse de la parole se divisent en deux grandes familles, les synthétiseurs à partir d'un concept et les synthétiseurs à partir du texte. La première famille s'intègre dans les systèmes de dialogue homme-machine et génère la parole à partir de l'état pragmatique et sémantique du discours. La deuxième, qui nous concerne, reçoit un texte orthographique en entrée, provient des sources différentes (clavier, Internet ... etc.), et son objectif est de produire la prononciation de ce texte sans aucune information sur la nature de ses mots (sigle, abréviation, chiffre, date, etc.). Un système TTS est généralement composé de deux blocs. Un pour le traitement linguistique ou Traitement de Langage Naturel (TLN) « bloc de haut niveau », et un bloc de traitement acoustique « bloc de bas niveau » appelé aussi bloc de TAP [1, 14, 22, 23] (figure 1.14).

1.4.4.1 Traitement linguistique

Cette étape permet le passage de la forme textuelle (représentation orthographique) à une représentation phonétique munie d'une description prosodique de la phrase à synthétiser. Les principales étapes de ce bloc sont [24] :

- normalisation et prétraitement du texte brut : elle consiste à bien déterminer les phrases à synthétiser et interpréter les caractères spéciaux, abréviations, chiffres, etc. D'une autre façon, c'est la conversion du texte brut en une séquence d'unités lexicales ou mots ;

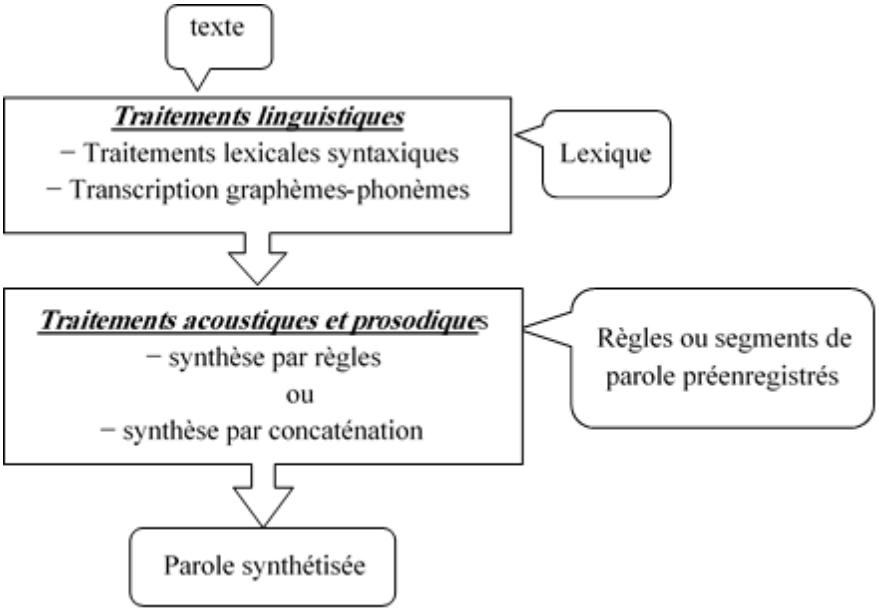

Figure 1.14: Schéma synoptique d'un synthétiseur à partir du texte [24]

- analyse lexicale, morpho-syntaxique et syntaxique : dans cette étape, la séquence résultante de la dernière étape est enrichie par des étiquettes lexicales et des marques morpho-syntaxiques, pour produire à la fin un découpage en composantes syntaxiques ;
- transcription graphème-phonème : C'est la conversion de la suite lexicale de sa forme orthographique à la forme phonétique.

1.4.4.2 Traitement acoustique et prosodique

Le deuxième bloc d'un système TTS a pour objective de transformer la suite des symboles obtenu de la première phase en une suite de segments acoustiques qui confondent à la prosodie cible. Les modules constituant cette étape dépondent de la méthode utilisée (paramétrique ou par concaténation).

1.4.5 Défis et problèmes dans la synthèse de la parole

Les défis et problèmes confrontés dans la construction d'un système de synthèse peuvent se classer selon le bloc de synthèse étudié [17].

1.4.5.1 Dans le haut niveau

Le premier défi dans les systèmes TTS réside dans la conversion du texte à une représentation linguistique de sa prononciation, appelée la Transcription Orthographique Phonétique (TOP). La difficulté de celle-ci varie d'une langue à une autre. Dans certaines langues comme le Finlandais, où le texte presque correspond

à sa prononciation, la TOP peut être considérée comme simple. Tandis que, dans autres langues comme le Français et l'Anglais la transcription est un peu plus compliquée, car elle nécessite un grand nombre de règles et d'exceptions pour une conversion correcte. Concernant l'Arabe, l'absence des signes diacritiques dans les textes d'utilisation générale rend la prononciation du mot ambigu et la transcription encore plus difficile. Les difficultés de transcription englobent aussi les défis dans le prétraitement du texte qui concerne la bonne interprétation des chiffres, symboles et abréviations ; par exemple : 19/5 peut s'interpréter comme une fraction ou comme une date, l'abréviation « St » en anglais peut signifier saint ou street, etc.
Un autre grand problème concerne ce bloc est de trouver la bonne valeur d'intonation, de durée et de stress pour chaque mot de la phrase à synthétiser (les caractéristiques prosodiques ou suprasegmentales). Car ces propriétés ne peuvent pas être extraites du texte mais du sens de la phrase et les caractéristiques du locuteur et ses émotions.

1.4.5.2 Dans le bas niveau

Les problèmes et défis comprises dans le module de traitement acoustique reflètent les inconvénients des méthodes et techniques adoptées. Dans la synthèse articulatoire, la collection des données et l'implémentation des règles pour les commander est une tâche très complexe. En plus, il est presque impossible de modéliser parfaitement le système phonatoire humain (les cordes vocales, les mouvements de la longue, etc.). Ce problème de la bonne modélisation et le grand nombre de règles nécessaires, se considère aussi un grand défi dans la synthèse par formant, notamment, les sons nasals. Dans la synthèse par concaténation, la collecte des segments sonores, et leur étiquetage prennent beaucoup de temps. Le ressoude du problème de la distorsion et la discontinuité créé dans les points de concaténation est aussi un grand défi dans cette méthode.

1.4.6 Evaluation des systèmes de synthèse de la parole

L'évaluation d'un système de synthèse est une étape très importante dans son développement, mais elle peut être une tâche difficile, fastidieuse et parfois coûteuse. Son problème principal se pose dans le choix des tests à effectuer parce qu'il y a plusieurs méthodologies d'évaluation. Cette dernière peut être subjective ou objective, se fait par diagnostic ou comparaison, par module ou globalement. De plus, elle peut être effectuée avec des haut-parleurs ou casques, par des spécialistes ou des personnes

naïves, dans un milieu normal ou isolant. Toutes ces variantes ont poussé certains chercheurs à inquiéter, car elles mettent la comparaison et la standardisation des systèmes de synthèse difficile. Pour avoir une meilleure évaluation, il est conseillé de choisir quelques méthodes pour évaluer des caractéristiques séparées, et correspond aux besoins de l'application [21, 22, 25].

1.4.6.1 Méthodes d'évaluation

De point de vue méthodologie, l'évaluation d'un système de synthèse peut être subjective ou objective. Entre ces deux, les tests objectifs peuvent être préférable du premier coup dû au coût réduit et leur rapidité, mais ils ne peuvent pas remplacer l'évaluation subjective. Malgré que cette dernière nécessite un grand nombre d'exemples de test et d'auditeurs, nous ne pouvons pas négliger que les systèmes de synthèse sont désignés à l'utilisation humaine.

- *tests subjectifs :* l'évaluation subjective d'un système de synthèse est une mesure de qualité de la voix par un groupe d'auditeurs, ou ils basent sur leur perception pour classer cette qualité dans une échelle bien déterminée. Dans la plupart des cas, les résultats de cette évaluation engendrent un certain degré de variation. Pour cela, il faut moyenner les résultats de certains nombres de personnes et éliminer ceux qui sont très loin. Parmi les méthodes d'évaluation subjectives utilisées, nous distinguons les tests de préférence comme le AB et ABX ; et les tests de score comme : le MOS (Mean Opinion Score) le DMOS (Degradation Mean Openion Score). Pour une meilleure évaluation, la plupart des recherches estiment que le nombre d'évaluateurs doit au minimum entre 20 et 30. Tandis que les phrases de test doivent être choisies dans différents contextes et domaines ;

- *tests objectifs :* plusieurs mesures objectives ont été proposées afin d'évaluer la parole synthétique avec de bons critères. Généralement, ces tests se basent sur la mesure de la dégradation du signal synthétisé, par rapport à une version originale (naturel). Parmi ces mesures nous notons :

  - le SNR (Signal to Noise Ratio) et le SNR_{seg} (par segment) sont le rapport du signal synthétisé sur la différence entre le signal original et synthétisé ;
  - le PESQ (Perceptual Evaluation of Speech Quality) est un algorithme principalement conçu pour l'évaluation objective de la qualité d'un canal

de transmission ou d'une opération de codage de la parole. Il se fait par la comparaison du signal résultant avec une version de référence. Ce test est utilisé pour l'évaluation de la parole synthétisée après que Cernak et Rusko (2005) ont prouvé la corrélation entre leurs résultats par le PESQ et ceux d'un test MOS.

Cependant, ces méthodes ne sont pas suffisamment fiables, notamment pour les systèmes de synthèse par sélection d'unités, la dégradation entre un signal original et un autre synthétisé ne signifie pas que ce dernier est de mauvaise qualité. Pour cela, certains travaux ont été faits pour prédire la présence et la position des artéfacts dans la synthèse par sélection d'unité comme une évaluation objective.

1.4.6.2 Evaluation de la qualité de la parole

Il existe plusieurs façons pour évaluer la qualité d'une parole synthétique. Nous pouvons la tester selon son intelligibilité, naturelle ou par rapport son aptitude à l'application visée (compréhensivité, rapidité, expressivité ..., etc.). Par exemple, les lecteurs de texte automatique pour les personnes non voyantes nécessitent en général plus d'intelligibilité avec un débit de parole élevé, cependant les applications dans le multimédia c'est le naturel de la parole et certaines propriétés prosodiques qui sont importantes.

- *l'intelligibilité :* l'évaluation de l'intelligibilité de la parole est généralement faite par des tests d'écoute subjectifs. Ces derniers s'appliquent selon plusieurs niveaux (phonémique, mot, phrase) et dépendent de l'information désirée et du but de l'évaluation. Par exemple, dans une conversation ordinaire, il peut être acceptable de ne pas reconnaître quelques mots si le message global est compréhensible. Un premier type de ces tests est de présenter aux auditeurs un groupe de mots et nous leur demandons de les transcrire. Ces mots peuvent se différencier dans la première consonne seulement (Diagnostic Rhyme Test (DRT)), ou bien dans la première et dernière consonne seulement (Modified Rhyme Test (MRT)). À la fin du test, le taux d'erreur de la consonne est calculé pour chaque mot et la meilleure performance c'est celle du taux le plus faible. L'autre type de tests consiste à demander aux auditeurs de transcrire un groupe de phrases, où nous testons l'intelligibilité des mots dans le contexte des phrases. Ces phrases peuvent être avec sens, comme le test « Harvard Psychoacoustic

Sentences (HPS) », ou bien sémantiquement fausse comme dans les tests « Haskins Syntactic Sentences » ou SAM Semantically Unpredictable Sentences (SUS). Ces deux derniers tests sont un peu difficiles pour les auditeurs, car dans le cas où il y aurait des mots non intelligibles, nous ne pouvons pas les prédire à partir du contexte de la phrase ;

- *le naturel :* l'évaluation du naturel est une mesure de la ressemblance entre la parole synthétisée et une voix réelle. Les tests utilisés pour cette qualité de parole sont généralement des mesures subjectives où plusieurs paramètres peuvent influer sur ce jugement. Par exemple, il faut clarifier aux évaluateurs qu'ils doivent juger le réalisme de la parole synthétisée et non pas la qualité globale. Le MOS est le test le plus simple pour évaluer la qualité de la parole en général, et plus précisément le naturel. C'est une classification à une échelle de cinq niveaux, où trois types d'évaluations sont principalement utilisés. Premièrement, l'évaluation absolue (Absolute Category Rating ACR) consiste à noter la qualité des phrases de tests de 1 (très non naturelle) à 5(très naturelle). Deuxièmement, l'évaluation de la dégradation (Degradation Category Rating DCR), où nous présentons aux auditeurs deux échantillons de parole pour chaque phrase, l'une synthétisée et l'autre réelle. Dans ce cas-là, ils doivent noter le niveau de « dégradation » de la parole synthétisée par rapport au celle naturelle (réelle). Finalement, l'évaluation comparative (Comparison Category Rating CCR) permet de comparer les performances de plusieurs systèmes de synthèse. Les échantillons sont présentés par paires et dans un ordre aléatoire, et l'évaluateur doit noter la supériorité d'un système par rapport à l'autre. À la fin de test les résultats vont se moyenner et le score final est déterminé. En plus de l'examen d'intelligibilité et du naturel de la parole synthétique, nous pouvons ajouter des tests de compréhension des phrases. Dans ces derniers si les auditeurs répondent correctement aux questions concerne le contenu du paragraphe, le système est validé même s'il y a quelques phonèmes ou mots mal synthétisés. Dans d'autres applications spécifiques, nous pouvons aussi évaluer la plaisance ou l'expressivité de la parole.

1.5 Conclusion

Ce chapitre nous a donné une vue globale sur le domaine de la parole et ses caractéristiques sur le plan physiologique, acoustique, phonologique et phonétique. Après une brève description de la langue arabe. Nous avons ensuite familiarisé avec le domaine de synthèse de parole, ces principales techniques et méthodes. Nous avons aussi vu que l'évaluation d'un système de synthèse est une tâche fatigante, ennuyante qui nécessite beaucoup d'études et préparations. D'après ce chapitre nous estimons que la synthèse par sélection d'unité est préférable pour notre application à cause de la qualité de la parole qu'elle peut produire. Cette méthode va être décrite en détail dans le chapitre suivant ainsi que certains outils utilisés pour la construction de notre système de synthèse.

CHAPITRE 2 : SYNTHÈSE PAR SÉLECTION D'UNITÉS, NOTIONS SUR LES SYSTÈMES EXPERTS ET LES ALGORITHMES GÉNÉTIQUES

2.1 Introduction

Le travail de cette thèse vise à élaborer un système de synthèse à partir du texte. D'après le chapitre précédant, la Synthèse par Sélection d'Unités (SSU) acoustiques est l'une des méthodes préférées à cause de sa qualité de parole naturelle. Pour cela, nous allons décrire de ce chapitre cette méthode de synthèse ainsi que quelques outils nécessaires pour son développement. Nous commencerons par l'historique de la SSU et son principe de fonctionnement, pour arriver à comment un système de SSU peut être construit. Après nous présenterons des notions fondamentales sur des techniques utilisées pour l'amélioration du système de synthèse. Elles consistent des systèmes experts et les Algorithmes Génétiques.

2.2 Synthèse par Sélection d'Unités acoustiques

La SSU est une méthode de synthèse par corpus. Elle se base sur la concaténation des meilleurs segments sonores confondent (linguistiquement et prosodiquement) avec le texte désiré. Ces segments s'appellent aussi unités, doivent être bien sélectionné à partir d'une grande BD riche d'unités de différentes tailles dans des contextes prosodiques et linguistiques distincts. Suivant le principe « sélectionner le meilleur et modifier le moins », cette méthode réside sur l'enrichissement de la BD pour minimiser ou annuler carrément, le besoin du traitement de signal (avec PSOLA par exemple) qui existe dans d'autres techniques de synthèse par concaténation. Car toute modification des caractéristiques de signal affecte le naturel de la parole.

2.2.1 Algorithme de Black et Hunt

Les premières recherches sur la SSU ont commencé à la fin des années 80 par Sagisaka et Hirokawa [26, 27]. Ils initient de l'idée d'agrandir la BD par une variante d'unités acoustiques avec le défi de sélectionner les meilleures de manière optimale. Cependant, c'est en 1996 où les premiers principes de cette méthode ont été établis par Black et Hunt [28] avec leur système « *CHATR* » qui utilise le diphone comme unité de base. L'algorithme de Black et Hunt est défini comme étant une recherche dans la BD (structuré comme un réseau de transition d'état de diphones), pour trouver la meilleure séquence d'unités $\widehat{U}$. Cette recherche suit l'algorithme de Viterbi, et comme indique l'équation (2.1), la sélection de ces unités base sur le calcule et la minimisation d'une fonction de coût comporte deux coûts élémentaires. Le calcul de ces derniers est

présenté dans les équations (2.2) et (2.3). Ils désignent, premièrement, le coût cible (C_t) qui évalue le pourcentage de ressemblance entre les unités cibles $T =< t_1, t_2, ..t_N >$, issues du texte entré, et les unités candidates dans la BD : $U < u_1, u_2, .., u_N >$. Et deuxièmement, le coût de concaténation (C_c) qui reflète le pourcentage d'adaptation de chaque unité avec celle adjacente (une valeur minimale => une bonne jonction), (figure 2.1) [16, 29].

$$\widehat{U} = \min_{U_1,..,U_N} \left\{ w_1 (\sum_{i=1}^{n} C_t(t_i, u_i)) + w_2 (\sum_{i=1}^{n} C_c(u_{i-1}, u_i) + C_c(S, u_1) + C_c(u_n, S)) \right\} \quad (2.1)$$

$$C_t(t_i, u_i) = \sum_{j=1}^{q} w_j^t C_j^t(t_i, u_i) \quad (2.2)$$

$$C_c(u_{i-1}, u_i) = \sum_{k=1}^{p} w_k^c C_k^c(u_{i-1}, u_i) \quad (2.3)$$

où :

n est le nombre des unités de la phrase à synthétiser ;

$C_t(t_i, u_i)$ est le coût cible entre l'unité désirée t_i et la candidate u_i ;

$C_c(u_{i-1}, u_i)$ est le coût de concaténation entre les unités successives u_{i-1}) et u_i ;

S représente le silence,où $C_c(S, u_1)$ et $C_c(u_n, S)$ définissent le coût de la première et la dernière unités avec le silence ;

q et p représentent respectivement le nombre de caractéristiques utilisées pour le coût de concaténation et le coût cible ;

w_j^t et w_k^c sont les poids de la j^{ème} caractéristique du coût cible et la k^{ème} caractéristique du coût de concaténation.

2.2.2 Base de données / Corpus

En plus de l'appellation Base de Données (BD), nous entendons souvent le mot corpus. Ils sont deux termes avec un sens conjoint dans le domaine de synthèse de la parole. Une BD est une collection de variants types de données comme nous l'on besoin dans un système de synthèse. Tandis qu'un corpus est une grande collection de textes. Il s'agit d'un ensemble de documents écrits ou parlés sur lesquels repose une analyse linguistique [30].

2.2.2.1 Corpus

Afin d'élaborer une bonne BD pour un système de SSU, un corpus doit être bien étudié et établi. Ce dernier se représente comme un grand enregistrement de parole

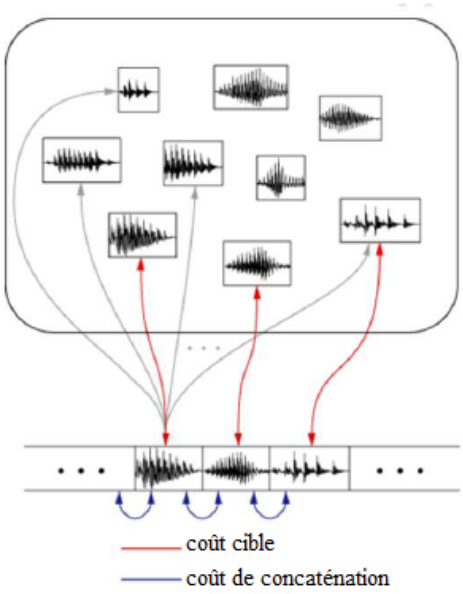

Figure 2.1: Illustration de calcul des fonctions de coût [5]

équilibrée phonétiquement, linguistiquement et acoustiquement. Le choix de ce qu'il faut y inclure ou non n'est pas évident du tout. La taille du corpus, sa qualité d'encodage, le locuteur utilisé pour l'enregistrement et le genre littéraire de texte à employer doivent tous mettre en jeu. Il existe deux principales méthodes pour la préparation d'un corpus. La première dit par condensation, où nous supposons l'existence d'un grand corpus, riche phonétiquement et linguistiquement mais très énorme à enregistrer. Le nouveau corpus que nous aurons le préparer dans ce cas-là est une version condensée du premier, où nous essayons de réduire sa taille par un choix des « n » énoncés (généralement phrases) comprennent les attributs les plus intéressantes. Ceci qui n'est pas facile à faire, car nous devons faire un compromis entre la taille du corpus et sa couverture. Dans la deuxième méthode le corpus se prépare par la détermination des phrases couvrantes les spécifications de l'application désirée. Le problème de cette méthode c'est que nous tombons des fois sur des phrases sans aucun sens dans la langue qui mène à une difficulté de prononciation pour la personne qui va faire l'enregistrement [25].

2.2.2.2 Taille d'un corpus / BD

Peu importe l'unité de base utilisée, le corpus doit contenir plusieurs heures de parole neutre. Il doit être varié et assez riche en termes d'unités phonologiques actuelles dans la langue de synthèse pour garantir que l'algorithme de sélection trouvera les unités suffisamment proches aux celle cibles, par conséquent une qualité de parole synthétique meilleure. Mais le problème qui se pose dans ce cas est la mémoire de stockage qui peut être énorme. De plus, la qualité de codage peut jouer un rôle très

important. Généralement la taille d'un corpus dépend de deux éléments [25] :

- le but du système de synthèse, où la question principale qui se pose est : est-ce que ce système sera utilisé dans un domaine spécifique, générale ou entre les deux. Par exemple, un système « Global Positioning System (GPS) » comprend de nombreuses phrases / mots préenregistrés du vocabulaire de conduite comme : route, rond-point ou virage. Mais comme il peut aussi prononcer les noms des rues ou des villes, il doit générer des unités (mots) plus général qui peuvent se couvrir par l'utilisation des diphones. Le public ciblé peut aussi influencer la taille du corpus. Car par exemple un système qui lit les livres destinés aux enfants ne sera probablement pas adapté à la même tâche pour les adultes;

- les contraintes techniques de la plate-forme où le système de synthèse devrait être utilisé, car nous pouvons pas agrandir un corpus avec une faible plate-forme. Dans ce cas nous devrons vérifier :

  - si la parole synthétisée doit être générée en temps réel ;
  - la puissance de calcul, les vitesses de stockage et d'accès au « Random Access Memory (RAM) » du périphérique qui devra effectuer la tâche de synthèse ;
  - la capacité de stockage mémoire disponible sur le périphérique.

2.2.2.3 Qualité d'un corpus

La qualité d'un corpus, ou plus généralement BD dépend principalement de la qualité des sons utilisés. De plus, la bonne segmentation et annotation de celui-ci joue aussi un rôle très important. Cette dernière étape signifie la segmentation de la parole enregistrée au type d'unité adoptée, et la définition de leurs caractéristiques phonétiques, linguistiques et prosodiques. C'est une étape fastidieuse, consommatrice de temps et ressource humain.

2.2.2.4 Type des unités

Les anciens systèmes de synthèse par concaténation, adopte souvent le diphone comme unité de base. Cependant, dans la SSU l'utilisation du diphone peut nous limite, à cause de la variabilité exigée dans la SSU. De plus du variété contextuelle, linguistique et phonétique pour chaque unité, cette méthode offre le mélange des

types d'unités (les systèmes hétérogènes ou la synthèse des unités non uniformes). Le choix du type qui convient dépond de l'application ciblée et ses ressources (la mémoire de stockage disponible et la RAM par exemple) et la variété phonétique, linguistique ou acoustique qui peu couvre l'unité dans la langue. En plus des unités présentées dans le chapitre précédent (phonème, diphone, syllabe, mot et phrase) voici autres unités utilisées [16] :

- demi-phonème : une unité qui a la moitié de la taille du phonème. Elle s'étend soit de début du phonème à un point milieu, ou de ce point médian à la fin du phonème ;
- trame : une tranche du signal vocal;
- états : des partie du phonème, souvent déterminées par l'alignement des états HMM ;
- demi-syllabe : comme le demi-phonème cette unité reflète la moitié de la syllabe;
- di-syllabe : une unité qui s'étend de la moitié du premier syllabe au moitié du deuxième.

2.2.3 Sélection des unités

Le cœur d'un système de SSU est de trouver l'ensemble des unités acoustiques optimales à partir de la BD. Ceci, ce fait par la minimisation de la dégradation du naturel de la parole, causée par divers facteurs comme : la différence prosodique, la différence spectrale ou l'inadéquation des environnements phonétiques. Dans la littérature, nous trouvons que la méthode de sélection initiée par Black et Hunt [28] est devenu la base de la plupart des systèmes de SSU. Elle est préférable car elle ne limite pas le type ou la nature des caractéristiques des unités utilisées ni leurs valeurs. Comme déjà expliqué, elle se base sur le calcul et la minimisation de deux coûts, cible et concaténation, par la formulation des fonctions qui pénalisent les unités comme le ferait un être humain. La question principale que nous devons résoudre dans cette sélection est : quelles caractéristiques et critères devraient être évalués et quelle pondération devrait être opérée entre eux ? Bien sûr, la réponse de cette question dépend de la langue ciblée [25].

2.2.3.1 Présélection

La recherche des unités de différentes tailles dans un corpus est coûteuse en temps de calcul. À cet égard, une étape de prétraitement est souvent mise en œuvre pour diminuer le nombre des unités candidates et accélérer le processus de sélection. Cette étape appelée présélection, et son but est de réduire la taille du corpus par élimination des unités très différentes aux celles désirées. En général, deux types de présélection sont adoptés par clustering ou par filtres. Dans la première les unités similaires (phonétiquement, acoustiquement ou contextuellement) de la BD se regroupent en clusters et cette étape élimine les unités qui s'éloignent le plus des centres des clusters. Pour réaliser la présélection avec des filtres, une clé contenant des informations discrètes (principalement binaires) est créée pour chaque segment de la parole dans le corpus. Cela permet à l'algorithme de prendre ou de rejeter l'unité rapidement en comparant simplement les valeurs de cette clé avec les valeurs cibles. La clé peut contenir des informations phonétiques, linguistiques ou prosodiques. Voici un ensemble de filtres utilisés par Guennec [25] dans le système de synthèse « *IRISA TTS* » dans l'étape de présélection, pour chaque segment de la parole constituant une unité [16, 25] :

- est-il un son qui ne siffle pas ?
- est-ce qu'il est au début de la syllabe ?
- est-ce qu'il est dans la coda de la syllabe ?
- la syllabe actuelle est-elle à la fin du mot ?
- la syllabe actuelle est-elle dans le début un mot ?

Dans cet exemple, si aucune unité correspondante à l'ensemble des filtres n'est trouvée, les filtres de présélection sont relâchés un par un, à partir de la fin de la liste. Ce mécanisme permet de trouver un chemin dans tous les cas, mais l'inconvénient est que nous pouvons trouver des candidats loin des caractéristiques cibles que nous voulons, qui risque de produire des artefacts.

2.2.3.2 Fonction cible / coût cible

Le but de la fonction cible consiste à évaluer l'aptitude d'une unité à une spécification désirée, qui est fournie après le traitement du texte entré. Cette évaluation aboutit à une liste classée de toutes les unités de la BD, chacune avec un score, un coût ou une

distance, tout dépend des caractéristiques utilisées et le choix du développeur. En pratique, le calcul de la fonction cible est, généralement, appliqué après élimination des unités qui ne correspond pas aux spécifications désirées, soit par une bonne organisation de la BD, par clustering, ou par présélection. En d'autres termes, si la spécification désirée est celle du diphone [na], seules les unités dans la BD comprennent ce diphone sont considérées, c'est la détermination des listes des unités candidats. La taille de ces listes varie en fonction de l'unité et sa disponibilité dans la base et dans la langue. Elles peuvent comprennent quelques exemplaires jusqu'aux centaines d'unités. Avec la notion du coût cible qui est la plus adopté (elle peut être score où distance), la fonction cible se calcule en comparant les caractéristiques des unités cibles avec celles candidats (équation (2.2)). Comme nous avons déjà mentionné, le défi dans ce calcul est le bon choix des caractéristiques et leur nombre. Mais nous trouvons que peu de recherches détaillent ce problème car il est relié à la langue et l'application désirée. Cependant ce qui primordial est que ces caractéristiques sont définies avant la préparation du corpus. Car ce dernier est basé sur les caractéristiques que nous voulons adopter. Et puisqu'elles sont un choix personnel, elles doivent être bien définies pour être adaptées à l'application visée. D'une manière générale ces caractéristiques correspondent aux [16, 25] :

- descripteurs linguistiques et phonétiques des unités cibles (acquis par annotation automatique) ;
- valeurs prédites telles que F0, durée phonémique, énergie, etc ;
- règles (surtout prosodiques).

2.2.3.3 Fonction de concaténation / coût de concaténation

Le but de la fonction de concaténation consiste à donner dans quelle mesure deux unités se rejoignent lorsque se concatènent. Dans la plupart des approches, cette fonction renvoie un coût, appelé coût de concaténation. Ce dernier pénalise les jonctions susceptibles de provoquer un artefact de concaténation ou toute autre incohérence. Cependant, d'autres formulations sont aussi possibles, y compris le classificateur de jointure qui renvoie vrais ou faux, ou bien la probabilité de jointure qui renvoie la probabilité que deux unités seront trouvées en séquence. Empiriquement, une synthèse réalisée avec un coût de concaténation seulement est généralement acceptable. Alors que la synthèse faite qu'avec le coût cible aboutit

souvent à des phrases non intelligibles. On pourrait donc dire que le coût de concaténation est l'élément principal dans la sélection. De façon générale, les différences acoustiques et prosodiques entre les unités à joindre ont un impact sur le succès de la jointure, d'où l'utilisation de la distance dans le calcul de la fonction de coût. La façon la plus simple de mettre ceci en œuvre, est de comparer la dernière trame de l'unité gauche avec la première trame de l'unité droite. Comme la composition de la fonction du coût de concaténation est particulièrement importante, les chercheurs la donnent plus d'intérêt que la fonction du coût cible. Pour cela, de nombreuses mesures acoustiques et distances ont été implémentées pour les sous-coûts de concaténation. Nous notons les mesures de F0, l'énergie, les coefficients cepstrales, les formants, coefficients de prédiction linéaire, etc. Et parmi les distances D utilisées [16] :

- la distance de Manhattan

$$D = \sum_{i=0}^{N} |(x_i - y_i)| \tag{2.4}$$

- la distance euclidienne

$$D = \sqrt{\sum_{i=0}^{N} (x_i - y_i)^2} \tag{2.5}$$

- la distance mahalanobis

$$D = \sqrt{\sum_{i=0}^{N} (\frac{\frac{num}{x_i - y_i}}{\sigma_i})^2} \tag{2.6}$$

- la distance de Kullback–Leibler

$$D = \sum_{i=0}^{N} (x_i - y_i) \log \left(\frac{x_i}{y_i}\right) \tag{2.7}$$

tel que :

x et y sont les vecteurs qu'on veut calculer la distance entre eux ;

N la taille des vecteurs x et y ;

σ c'est l'écart type de x_i par rapport à la serie de données.

2.2.3.4 Pondération des coûts

Le problème le plus difficile dans la formulation des fonctions des coûts n'est pas de savoir quels sous-coûts devraient être utilisées mais quelle pondération doit être faite entre eux. Comme indiquent les équations (2.1), (2.2) et (2.3) nous distinguons deux types de pondération des sous-coûts [16, 25]:

- w_1 etw_2 sont des poids désignés pour équilibrer entre le coût cible et le coût de concaténation. Soit de favoriser l'un sur l'autre, ou pour les mettre en même grandeur si les coûts ne sont pas normalisés;

- w_j^t et w_k^c qui ont le même but que les précédents. Ils sont les poids donnés à chaque sous coût des fonctions cible et concaténation.

Ces poids peuvent être fixés dans le moteur TTS ou bien ils se misent à jour au cours de la synthèse, en fonction des caractéristiques des unités. La formation efficace de ces poids est un enjeu clé pour la meilleure sélection des unités candidates. Cependant, l'ajustement des poids dans les fonctions des coûts est l'un des problèmes les plus difficiles lors de la conception des systèmes de SSU. Cela est dû au fait qu'ils doivent refléter l'importance relative de chaque caractéristique utilisée dans la sélection et d'intégrer d'une manière ou d'une autre les préférences subjectives des auditeurs [16, 25].

La configuration des poids peut se fait de manière générale où nous désignons un poids uni pour chaque sous coût. Aussi, elle peut être définie pour chaque type ou groupe d'unités, ou bien contextuellement. Dans cette dernière, nous devons déterminer pour chaque unité dans un contexte son propre poids. Malgré la simplicité et la rapidité qui semblent dans les deux premières configurations, la dernière se considère la meilleure à cause de sa précision. Cependant, son problème est qu'elle est gourmande en temps de calcul et de ressources matérielles nécessaires pour entrainer les poids pour chaque élément. Les méthodes d'ajustement des poids peuvent être classées en deux catégories comme suit [31].

2.2.3.4.1 Ajustement objectif

C'est l'utilisation des techniques et mesures objectives pour trouver la meilleure pondération des coûts. Parmi les techniques utilisées nous distinguons :

- la Recherche dans l'Espace des Poids (REP) : cette technique discrétise, d'abord, l'espace de recherche du poids W, puis elle travaille sur l'espace fini résultant. Les poids optimaux sont obtenus après une analyse par synthèse exploitant toutes les configurations possibles des poids. La procédure commence par un choix d'une expression cible tirée du corpus de la parole. Ensuite, le processus de sélection des unités est exécuté pour obtenir une expression synthétisée pour chacune des configurations. Enfin, la meilleure séquence d'unités candidates

(et donc, la meilleure configuration des poids) est celle qui donne la distance minimale (généralement calculée comme une distance cepstral) par rapport à l'expression cible. Ce processus est répété pour toutes les expressions cibles et l'ensemble de poids le plus cohérent est choisi comme la solution finale. Pour maintenir une précision d'ajustement raisonnable nous devons augmenter le nombre de poids (élargir l'espace de recherche). Pour cette raison cette approche devient rapidement impossible à mettre en œuvre car elle tend à être computationnellement coûteux (le calcul augmente exponentiellement avec le nombre des poids [31, 32] ;

- la Régression Multi-Linéaire (RML) : cette approche repose sur la résolution d'une régression multilinéaire entre une mesure objective (généralement une distance cepstrale) et les sous-coûts pour une unité cible donnée. Elle était appliquée pour la pondération des sous-coûts cibles pour chaque phonème ou groupe de phonèmes, suivant les étapes suivantes : premièrement, pour chaque unité exemple de la BD ces étapes doivent s'effectuer [28] :

  – traiter l'unité d'exemple comme unité cible ;

  – calculer la différence acoustique (comme une mesure objective) entre cette cible et tous les autres cas contenant les mêmes phonèmes dans la BD ;

  – identifier l'ensemble des n meilleures correspondances à cette cible (n=20 par exemple) ;

  – déterminer les sous-coûts $C_t(t_i, u_i)$ pour la cible et ses meilleurs correspondantes.

  Ensuite, collecter les distances objectives et les sous-coûts cibles sur toutes les unités exemples et leurs « n » meilleurs correspondantes. Après, utiliser la régression linéaire pour prédire la distance objective par une pondération linéaire des t sous coûts cibles, tout en utilisant les poids déterminés par régression linéaire comme poids pour les sous-coûts cibles, pour l'ensemble de phonèmes actuel. Enfin ces étapes se répètent pour chaque phonème ou groupe de phonèmes pour avoir les configurations finales. L'objectif de cet algorithme d'entraînement est de déterminer les poids pour les sous-coûts cibles qui sélectionnent des unités proches de ceux qui seraient sélectionnés si la mesure objective est utilisée directement dans la sélection de l'unité. La méthode de pondération des poids à régression présente de nombreux avantages par

rapport à la recherche d'espace de poids. En particulier, elle est capable de générer efficacement des poids séparés pour différentes classes de phonèmes, où l'influence du contexte prosodique et phonétique peut être différents. De plus elle est plus rapide car le temps d'entrainement est linéairement dépendant au nombre des sous-coûts contrairement au REP qui est exponentiel. Ce temps peut être réduit de 5 à 30 minutes si les distances acoustiques sont précalculées, ce qui permit une évaluation rapide des différents sous coûts ;

- les approches non linéaires : après la REP et la RML, plusieurs approches ont été proposées pour une pondération plus optimale. Parmi ces travaux, Park et al. [33], Kim et Park [34] proposent une technique de configuration des poids discriminante non linéaire. Dans cette dernière, le processus de sélection est considéré comme un problème de classification. Les poids sont mis à jour au moyen de la technique de descente en pente est l'erreur de classification est la mesure objective à optimiser. Elle était appliquée à l'ajustement des poids des sous coût cibles, laissant l'extension pour les poids de concaténation pour des travaux futurs. Dans le travail d'Alıas et Llorà [35] l'utilisation des Algorithmes Génétiques (AG) est introduite dans l'ajustement des poids pour les fonctions des coûts. Inspiré de l'idée d'évolution, les AGs font évoluer une population des poids candidats puis les adaptent à la fonction de coût de sélection d'unité. Ce processus prend l'avantage de mécanismes de la recombinaison de survivre le plus apte et du matériel génétique pour traiter les multiples des optima locaux de la fonction de coût, qui est une fonction non linéaire [36]. Les AGs peuvent surmonter les restrictions de REP et RML avec un calcul réalisable et l'utilisation d'une distance cepstrale pour évaluer la qualité de configuration des poids au sein de la fonction de fitness [35].

2.2.3.4.2 Ajustement subjectif

Ils sont des techniques qui servent à introduire la préférence humaine dans l'ajustement et l'entrainement des poids.

- l'approche manuelle : les pondérations des fonctions de coût peuvent être obtenues par un réglage manuel sous une supervision perceptuelle [37-40]. Ce processus commence par la formation d'un ensemble fini de poids utilisés pour synthétiser plusieurs énoncés (phrases de test). Ensuite, la configuration optimale des poids est déterminée après l'évaluation et le classement des

phrases synthétisées (généralement par des experts). Cette approche présente plusieurs problèmes. Par exemple, elle exige d'envisager un petit ensemble de poids pour rendre possible le processus de réglage. Comme proposé dans [41] les poids peuvent être choisis parmi 0.25, 0.5, 0,75, 1. En outre, le grand nombre des évaluations nécessaires pour ajuster les poids peuvent produire des résultats médiocres ou bruyants. En conséquence, l'ajustement manuellement des poids de la fonction de coût peut produire des résultats moins optimaux ;

- critère subjectif à priori : plusieurs travaux proposent différentes méthodologies pour établir une pondération optimale des sous coût, basant sur les tests de préférence perceptuelle comme étape préliminaire. Dans ces travaux, le MOS est utilisé car il est la mesure la plus acceptées pour évaluer le naturel de la parole synthétique subjectivement [38, 41, 42]. Cette approche se base sur la synthèse puis l'évaluation par MOS d'un ensemble de phrases utilisant différentes configurations des poids. Puis de trouver une corrélation entre les résultats MOS et les configurations des poids, dont les poids optimaux correspondent à la meilleure valeur de MOS ;

- les Algorithmes Génétiques interactifs (AGis) : sont utilisés dans la conception des fonctions de coût comme modèle d'optimisation des poids. Car ils sont capables de combiner l'ajustement non linéaire des paramètres quantitatifs et l'évaluation subjective des résultats, en remplaçant la mesure objective traditionnelle de la fonction de fitness par un processus de sélection basant sur la préférence humaine. Ce type d'algorithmes a été appliqué dans plusieurs disciplines pour fusionner la perception humaine et l'effort informatique lorsque l'évaluation subjective est un élément clé. Par exemple, l'ajustement perceptuel des appareils auditifs présenté dans le travail de Durant et al.[43], ou comme une technique de réglage des poids pour les systèmes de SSU dans Alías et al. [44]. Dans ce dernier, les résultats obtenus ont montré que les poids objectifs (obtenus par le RML et l'AG) étaient faiblement corrélés avec les poids subjectifs obtenus de la perception humaine. Cependant, les expériences ont mis deux principaux problèmes, qui sont l'ennui du processus causé par la fatigue de l'utilisateur et la complexité de maintenir le même critère de comparaison tout au long du processus (cohérence entre les utilisateurs) ce qui peut donner des résultats peu fiables.

Pour résoudre les problèmes causés par la AGi, surtout la fatigue de l'utilisateur, Alías et al. [31] adoptent un nouveau type des AG qui sont les Algorithmes Génétiques actifs interactifs (AGai). Comme présenté dans [45], les AGais ont montré leur progrès dans la lutte contre la fatigue des utilisateurs par la réduction du nombre d'évaluations en apprenant de l'interaction des utilisateurs et exploitant les connaissances acquises. Dans le problème de pondération des poids, les AGais font évoluer une population des poids, tout on les adapte à l'environnement donné (les préférences des utilisateurs). L'algorithme est conçu de telle sorte qu'il apprend de l'interaction avec les utilisateurs pour anticiper des hypothèses qui pourraient intéressantes l'ajouter dans la fonction fitness synthétisé. Cette méthode a été adaptée dans notre travail et comparée avec notre approche proposée (voire le chapitre suivant).

2.2.3.5 Procédure de sélection

La procédure de sélection des unités est un algorithme de recherche pour trouver la meilleure séquence finie des unités à concaténer. Ces dernières doivent être reliées et ordonnées entre eux avec un coût (poids) pour passer de l'une à l'autre. Donc la résolution de ce problème devient un processus de recherche du plus court chemin dans un graphe orienté pondéré $G = (V, A, C)$. Ce graphe représente la liste des unités candidats dans la BD, ses nœuds V sont les unités candidats, ses arcs A représentent les connexions possibles entre les unités. C sont les coûts de ces arcs, quantifiant le risque de créer des artefacts audibles lors de la concaténation de deux unités, comme représente la figure 2.2 [25]. L'algorithme le plus utilisé pour ce problème est de Viterbi. C'est un algorithme de recherche par programmation dynamique sous une forme de treillis des unités. Ce choix peut être justifié par sa recherche synchronisée dans le temps dans le graphe de sélection. Ceci est grâce à la propriété treillis du graphe de sélection qui permit une autre recherche inversée aussi. Comme présente la figure 2.3, premièrement, un treillis des unités candidats est formé en reliant les chemins entre les paires des nœuds possible (les unités candidats), représentés par des flèches en pointilles. Ensuite la recherche Viterbi se déplace de gauche à droite à travers ce treillis, tout en calculant un coût de chemin partiel (cumulatif). Ce dernier s'agit de la somme des coûts cible et de concaténation des unités dans un chemin donné. Au fur et à mesure que la recherche va plus loin, l'algorithme de Viterbi choisit et mémorise le meilleur chemin, jusqu'à présent

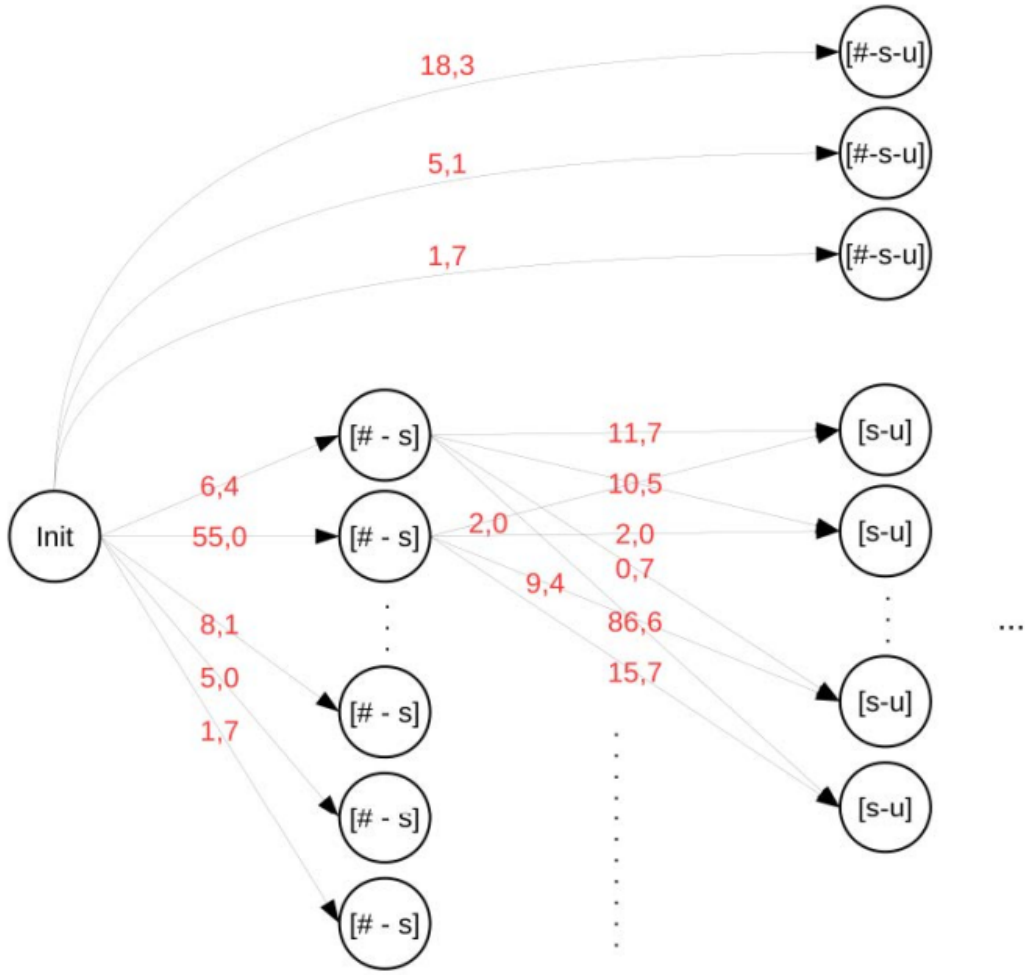

Figure 2.2: Exemple de modélisation en graphe du processus de sélection des unités [25]

(représenté dans la figure 2.3 par des flèches en gras). Par conséquent, pour une unité candidate donnée, l'algorithme ne prend que le chemin le moins coûteux jusqu'à ce point. Une fois la recherche est terminée, les unités formant le chemin avec le coût global le plus bas seront sélectionnées [46]. Pour certaines applications cet algorithme est un peu long. Pour cela des techniques d'élagage (comme la présélection) sont souvent appliquées pour accélérer la recherche. L'élagage est un peu d'art : dans la recherche complète de Viterbi, notre but est de trouver la séquence d'unités avec le plus bas coût, et si nous éliminons l'une des séquences possibles, nous risquons d'éliminer le meilleur chemin. Avec un peu d'habileté, de connaissances et de jugement, il est souvent possible de configurer une recherche qui considère moins d'unités, mais elle garde la bonne séquence en elle. Enfin, ce n'est pas toujours un désastre si le meilleur chemin est manqué, car ceci dépend s'il y a beaucoup de chemins près du score du meilleur ou non. Cela dépend, surtout, du nombre des unités les plus correspondantes à la spécification. Sinon, nous pouvons avoir seulement quelques unités un peu proches et de mauvais résultats si elles ne sont pas

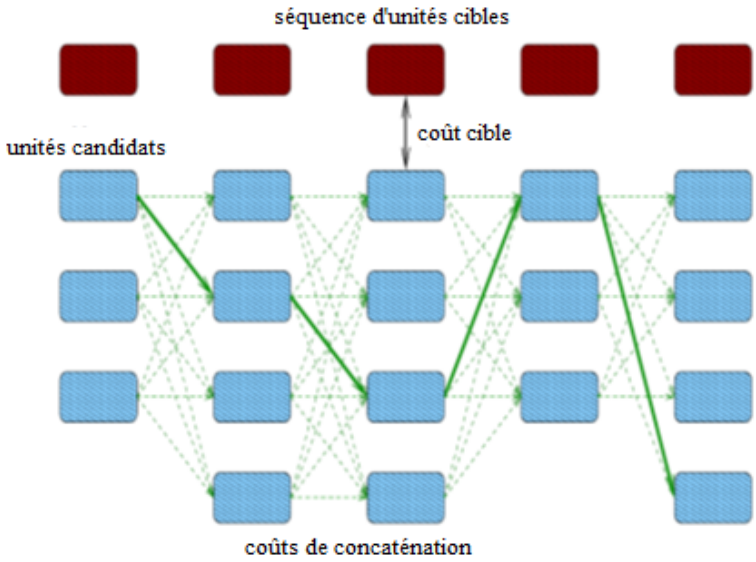

Figure 2.3: Structure des unités candidates dans la recherche de Viterbi [47]

prises en compte dans la recherche. Il convient toutefois de souligner qu'il n'y a pas de solution miracle à l'élagage. Si nous retirons des d'unités il y a toujours un risque que la meilleure séquence d'unité ne soit pas trouvée, et dans les cas des unités de base de la langue avec des caractéristiques rares, les conséquences peuvent être graves [16].

Un autre algorithme utilisé par Gunnec [25] est le A*. Gunnec a implémenté cet algorithme dans le but qu'on peut abouti à une séquence d'unités moins optimales mais acceptable avec moins de calcul. Contrairement à l'algorithme de Viterbi, qui fonctionne en réseau treillis contenant tous les nœuds candidats, le A* forme un graphe. À chaque instant, il explore le meilleur nœud du graphe en utilisant une fonction de coût $f(n)$ qui dépend du chemin à partir du nœud source $g(n)$ et le coût estimé pour arriver à la cible $h(n)$.

$$f(n) = g(n) + h(n) \tag{2.8}$$

À chaque étape, cet algorithme prend le nœud le plus prometteur selon $f(n)$ et dépense son successeur jusqu'à ce que la fin soit atteinte. $h(n)$ est une heuristique qui permet d'accélérer l'algorithme. Tout en privilégiant les nœuds qui semblent d'être sélectionnés sur un chemin optimal par rapport à ceux qui ont un meilleur coût $g(n)$, mais peuvent conduire à des coûts plus élevés au plus tard [25].

2.2.4 Concaténation des unités

Dans un système de SSU pure avec une BD très riche, la dernière étape de synthèse consiste à mettre bout à bout (concaténer) les formes d'ondes des unités sélectionnées

(figure 2.4). Dans cet effet, certaines recherches, comme celle de Conkie et Isard [46], proposent d'effectuer une recherche du meilleur point de concaténation entre les deux unités. Ils montrent une méthode pour trouver les trames provoquant le minimum de désadaptation spectrale entre les unités [25].

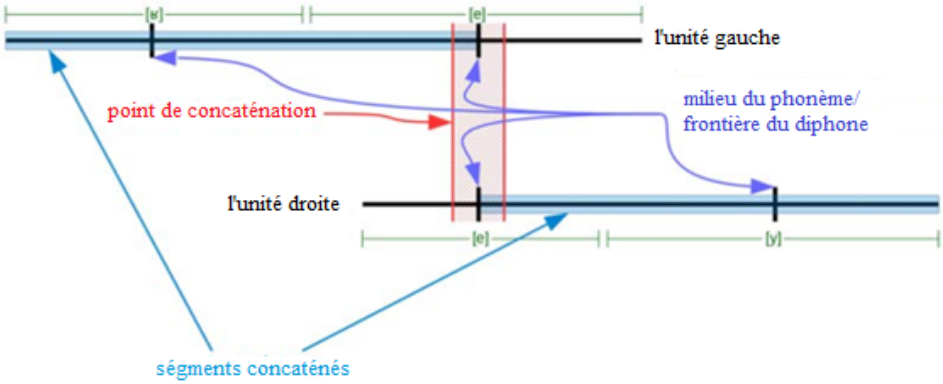

Figure 2.4: Schémas de concaténation des diphones français [ʁe] et [ey] [23]

Cette technique de concaténation fonctionne bien avec les applications à corpus, plus au moins, limité. Alors que dans les systèmes à utilisation générale, les discontinuités du son sont inévitables, car la richesse de BD est toujours limitée par l'espace mémoire disponible. Deux types de discontinuités peuvent exister lors de concaténation :

- distorsions de concaténation pure, causées, par exemple, par une interpolation ou lissage infructueux ;
- rupture prosodique, lorsqu'on joint des unités avec une prosodie très différente.

Pour résoudre ce problème, plusieurs techniques ont été testées pour éliminer ces discontinuités. Mais ils finissent par un étalement des artefacts sur des portions plus longues des segments concaténés.

La façon la plus courante de minimiser les distorsions est par une interpolation des segments pour les joindre sur quelques marks de pitch à la fin de l'unité gauche et au début de l'unité droite. Le segment interpolé est ensuite utilisé pour remplacer les parties originales. L'inconvénient de cette méthode est qu'elle est très basique et peut causer une rupture perceptible des trajectoires formantiques. Le problème de discontinuité a été aussi étudier par Pfitzinger [48] , ou il a proposé une solution en utilisant le lissage spectral. D'abord, les dérivées logarithmiques des spectres d'amplitude des deux signaux sont estimées. Puis les spectres sont alignés en utilisant « Dynamic Frequency Wrapping (DFW) », qui permet de calculer des réponses fréquentielles interpolées lisses (avec une interpolation linéaire pondérée entre les

deux représentations spectrales). Ce spectre est ensuite converti à des coefficients de filtre autorégressif réalisant une transition lisse entre les unités.

Pour les modifications prosodiques, les techniques comme PSOLA et ses versions, « Harmonic plus Noise Model (HNM) » ou STRAIGHT comme dans le travail de Kawahara et al. [49] peuvent effectuer des modifications prosodiques limitées de signal, principalement : l'adaptation du pitch et le réglage de la vitesse de parole. Cette modification doit rester modeste (pas plus de 1/5 de modulation fréquentielle de la parole par exemple) pour ne pas dégrader la parole générée. Cependant, l'adaptation prosodique est de moins en moins utilisée ces dernières années, pour deux raisons : premièrement, le risque de dégradation est important et deuxièmement, la synthèse paramétrique statistique permet un meilleur contrôle de la prosodie pour une bonne qualité de parole synthétique. La synthèse par modification de prosodie, ou n'importe quel traitement sur le signal de la parole produit des sons lisses et une qualité cohérente. Cependant, le naturel de la parole synthétique n'est souvent pas aussi bon que celle avec concaténation pure.

2.3 Systèmes Experts (SE)

Un Système Expert (SE), ou d'une manière générale un système à base de connaissances, est un programme qui simule le raisonnement d'un spécialiste dans un domaine précis. Le SE est un outil informatique qui comprend les connaissances et les compétences analytiques de plusieurs experts pour être utilisé par des non spécialistes. Il se base sur l'interactivité avec l'utilisateur par des questions et des réponses, pour lui permettre de prendre des décisions suite à des raisonnements. Le SE a remplacé l'expert humain et devient un outil de travail et d'aide dans plusieurs domaines comme le diagnostic (médical ou technique), la prévision, la classification, le dépannage, etc [50-52].

2.3.1 Architecture d'un système expert

Les principaux modules qui doivent contenir un système expert sont la base des connaissances et le moteur de raisonnement (d'inférence). Toutefois, autres composants peuvent être aussi existés [53, 54] (figure 2.5).

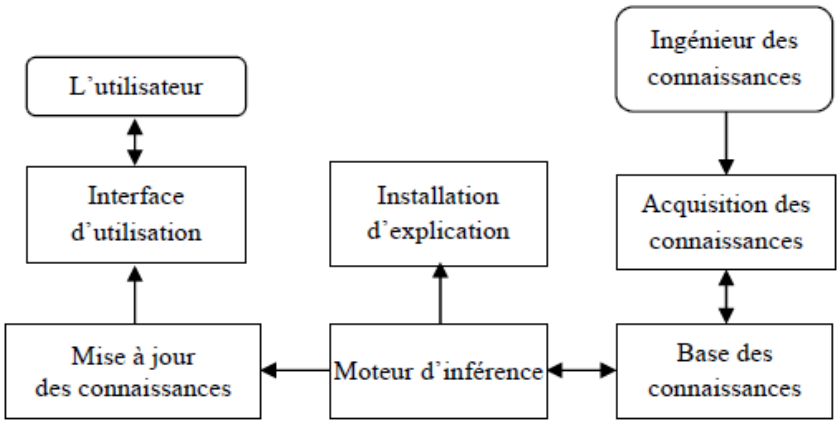

Figure 2.5: Eléments de composition d'un système expert [54]

2.3.1.1 Base des Connaissances (BCc)

La BCc contient les connaissances, propres au domaine, nécessaires pour comprendre, formuler et résoudre le problème posé. Elle est créée par l'ingénieur de connaissances, qui mène une série d'entretiens avec l'expert et organise les connaissances sous une forme facile à utiliser directement par le système. Elle comprend :

- la base de faits (expérience) qui sont des historisations et statistique des faits effectifs, des décisions et des buts. Le raisonnement va se baser sur ces faits pour déduire les conclusions ;
- les règles d'inférence (savoir-faire) qui sont les heuristiques de résolution du problème. Ils représentent les modes de raisonnement propres au domaine considéré.

La forme générale des connaissances est une représentation en formule d'inférence : si « antécédent », alors « conséquence », qui peut s'écrire sous forme de logique de premier ordre, règles de production ou en trame (objets) et réseaux sémantiques. Ces connaissances peuvent se représenter aussi sous forme de programmes conventionnels en langage haut niveau, dans le cas où les formules d'inférence nécessitent un calcul, une analyse ou interprétation pour être vérifiées ou validées.

2.3.1.2 Moteur d'Inférence (MI)

Le MI est le cerveau du système expert. Il agit comme un interpréteur qui analyse et traite les règles et fournit une méthodologie pour le raisonnement. La tâche principale du moteur d'inférence est de tracer son chemin à travers une forêt de règles pour atteindre un objectif précis. Ici, deux approches sont principalement utilisées.

- le chaînage en avant, qui est un processus d'inférence fondé sur les données (états initiaux). L'utilisateur du système doit fournir toutes les données disponibles, et le mécanisme d'inférence effectue une recherche dans les règles de la BCc jusqu'à ce qu'il trouve un antécédent (si « ... ») qui est connu pour être vrai. Une fois trouvé, il peut en déduire que la conséquence (alors « ... ») est vraie aussi, ce qui entraîne l'ajout de nouvelles informations à ses données. Ce processus se répète jusqu'à ce qu'un but soit atteint ou que plus aucune règle ne s'applique ;

- le chaînage arrière est un processus fondé sur les objectifs. Il commence par une liste d'hypothèses (objectives) et fonctionne à l'envers, de la conséquence à l'antécédent, pour voir s'il y a des données disponibles qui soutiennent l'une de ces conséquences. Par ce mécanisme le moteur d'inférence effectue une recherche sur les règles jusqu'à ce qu'il trouve celui qui a une conséquence qui correspond à un objectif désiré. Si l'antécédent de cette règle n'est pas connu pour être vrai, alors il est ajouté à la liste des objectifs. Le processus d'inférence s'arrêtera lorsque cette variable obtiendra une valeur ;

- une autre stratégie d'inférence existe aussi, qui est le mécanisme mixte, ou hybride. C'est une combinaison des chaînages avant et arrière. Le mécanisme de processus arrière est approprié, lorsqu'il y a peu d'états d'objectifs et de nombreux états initiaux. Tandis que l'enchaînement avant est préféré quand il y a peu d'états initiaux et de nombreux états d'objectif.

2.3.1.3 Interface d'Utilisation (IU)

L'IU est le moyen de communication avec l'utilisateur. Elle comporte sous forme d'un menu, interface graphique ou autre, pour mettre le dialogue et l'interactivité avec l'utilisateur convivial. Le but de cette interface est de convertir les connaissances (règles) de leur représentation internet (qui peut être compliquée) a une forme compréhensible par utilisateur.

2.3.1.4 Installation d'explications

C'est un module conçu pour expliquer les actions du SE. Ces explications peuvent être des réponses à comment le SE est arrivé aux solutions finales ou intermédiaires ou bien des justifications au besoin des données additionnelles.

2.3.1.5 L'acquisition de la connaissance

C'est l'assemblage, et la transformation de l'expertise du savoir résoudre le problème de l'expert du domaine ou un document source, à un programme pour construire la base des connaissances.

2.3.1.6 L'ingénieur des connaissances

Il est le constructeur du système expert. Il est responsable du développement du moteur d'inférence et de structurer l'interface utilisateur et la base des connaissances, ainsi que la maintenance de tous les modules. Cet ingénieur doit avoir les connaissances de la technologie des SE et devrait savoir comment développer un tel système. Il n'est pas nécessaire que l'ingénieur soit compétent dans le domaine dans lequel le système expert est développé. Mais une connaissance générale et la familiarité avec les termes clés utilisés dans le domaine sont toujours souhaitables, car cela permettra non seulement de mieux comprendre les connaissances du domaine, mais aussi réduire l'écart de communication entre l'ingénieur de connaissances et l'expert.

2.4 Algorithme Génétique (AG)

Les Algorithmes Génétiques (AGs) sont des méthodes de recherche stochastiques reposant sur les mécanismes de la sélection naturelle, de la génétique et le principe darwinien de survie du plus robuste. Ils sont introduits en 1975 par John Holland dans son livre « Adaptation in natural and artificial systems » [55]. Les AG sont l'une des techniques de calcul évolutionnaire, utilisés dans les problèmes d'optimisation complexes lorsque les méthodes classiques ou déterministes ne sont pas applicables ou ont échoué. De façon générale, les AGs sont une recherche évolutionnaire et cyclique d'une valeur optimale d'un problème donné dans un espace de solutions aléatoires [10, 56-58].

2.4.1 Principe de fonctionnement

Le principe général du fonctionnement d'un algorithme génétique est représenté dans la figure 2.6. La première étape consiste à générer une population initiale d'individus de façon aléatoire. Puis, trois opérations principales se répètent pour passer d'une génération k à une génération k+1 jusqu'à la satisfaction de la condition

d'arrêt. Des couples de parents P1 et P2 sont sélectionnés en fonction de leurs adaptations avec la solution du problème à résoudre, (par une fonction dite de fitness f(x)). Puis, l'opérateur de croisement est appliqué sur P1 et P2 avec une probabilité Pc (généralement autour de 0.6) qui va engendrer des couples d'enfants C1 et C2. Ensuite, d'autres éléments P sont sélectionnés en fonction de leur adaptation, et un opérateur, dit de mutation, est appliqué avec la probabilité Pm (elle est généralement très inférieure à Pc), ceci va engendrer des individus mutés P'. Les enfants (C1, C2) et les individus mutés P' sont ensuite évalués avant insertion dans la nouvelle population. Différents critères peuvent être choisis comme condition d'arrêt [56] :

- le nombre de générations que l'on souhaite exécuter, qui peut être fixé a priori. C'est ce qu'on est tenté de faire lorsqu'on doit trouver une solution dans un temps limité ;
- l'algorithme peut être arrêté lorsque la population n'évolue plus ou plus suffisamment rapidement (ceci veut dire que l'évaluation des individus par f(x) presque ne change plus).

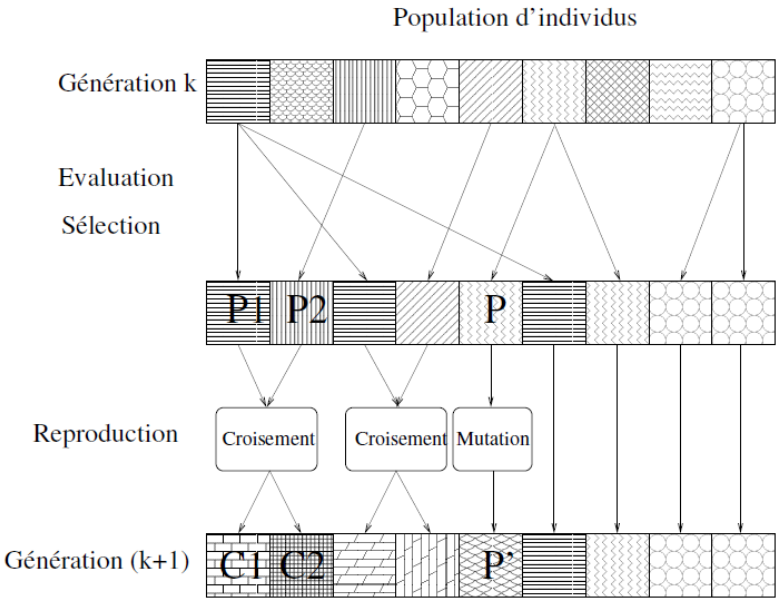

Figure 2.6: Principe des algorithmes génétiques [56]

2.4.2 Eléments principales

Comme expliqué dans le fonctionnement des AGs , ils reposent, principalement, sur [10, 56-58].

2.4.2.1 Codage des données

La qualité de codage des individus de la population joue un rôle très important dans le succès de la recherche de l'optimum. Le premier code utilisé dans les AGs est le binaire ordinaire, où chaque individu est représenté sous forme de chaînes de bits contenant toute l'information nécessaire à la description d'un point dans l'espace d'état. Ce type de codage est adopté pour simplifier les opérations de croisement et de mutation. Cependant, il n'est pas toujours bon. Deux éléments voisins en termes de distance de Hamming ne codent pas nécessairement deux éléments proches dans l'espace de recherche. Cet inconvénient peut être évité en utilisant un codage de Gray. Pour des problèmes d'optimisation dans des espaces de grandes dimensions, la structure du problème peut être altérée et le codage binaire devenu mauvais. Pour cela, les nouveaux algorithmes génétiques utilisent un codage réel : Real Coded Genetic Algorithms (RCGA) qui évite ce genre de problème et permet l'optimisation à grande dimension.

2.4.2.2 Population initiale

Le choix de la population initiale d'individus conditionne la rapidité de l'algorithme. Si la position de l'optimum dans l'espace d'état est inconnue, le choix des individus se fait aléatoirement. Il se fait par un tirage uniforme dans chacun des domaines associés aux composantes de l'espace d'état, en veillant à ce que les individus produits respectent les contraintes. Si par contre, des informations à priori sur le problème sont disponibles, il paraît bien à engendrer les individus dans un sous-domaine particulier afin d'accélérer la convergence de l'algorithme. Comme exemple, on considère une population composée de n individus. On veut optimiser m variables $(x_1^1\ ;\ x_2^1\ ;\ ...\ ;\ x_m^1)$, donc chaque chromosome aura m gènes. Donc la population est :

$$population = \begin{cases} I_1 = (x_1^1\ ;\ x_2^1\ ;\ \cdots\ ;\ x_m^1) \\ I_2 = (x_1^2\ ;\ x_2^2\ ;\ \cdots\ ;\ x_m^2) \\ \vdots \\ I_1 = (x_1^n\ ;\ x_2^n\ ;\ \cdots\ ;\ x_m^n) \end{cases} \tag{2.9}$$

Chaque individu Ii est composé d'un chromosome à m gènes dont les valeurs sont choisies aléatoirement. Normalement, chaque paramètre d'un individu a sa valeur propre (puisqu'on crée les individus de façon aléatoire), c'est-à-dire, $x_i^1 \neq x_i^2 \neq \cdots \neq x_i^n$ où $i = [1, 2, \cdots, m]$.

2.4.2.3 Gestion des contraintes

Un élément de population qui viole une contrainte se verra attribuer un mauvais fitness et fort probablement il va être éliminé par le processus de sélection. Cependant, il peut être intéressant de conserver ces éléments non admissibles, tout on les pénalise. Puisqu'ils peuvent permettre de générer autres éléments admissibles de bonne qualité. Gérer ça, n'est pas assez simple. Un compromis entre la favorisation des éléments admissibles et la pénalisation des autres doit être fait. Car la diversité de la population doit être entretenue au cours des générations, afin de parcourir le plus largement possible de l'espace d'état.

2.4.2.4 Opérateur de croisement

Le croisement a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes. Classiquement, les croisements se font avec deux parents et génèrent deux enfants. Initialement, le croisement associé au codage par chaînes de bits est le croisement à découpage de chromosomes. Pour effectuer celui-ci sur des chromosomes constitués de m gènes, on tire aléatoirement une position dans chacun des parents. Ensuite, on échange les deux sous-chaînes terminales de chacun des deux chromosomes, ce qui produit deux enfants C1 et C2 (figure 2.7 a). Ce principe peut être étendu en découpant les chromosomes non pas en deux sous-chaînes mais en trois, quatre, etc. (figure 2.7 b). Ce type de croisement est très efficace pour les problèmes discrets. Pour les problèmes continus, un croisement « barycentrique » est souvent utilisé. Soit deux gènes P1(i) et P2(i) sélectionnés dans chacun des parents à la même position i, les nouveaux gènes C1(i) et C2(i) sont définis par combinaison linéaire comme suit :

$$C1(i) = \alpha P1(i) + (1 - \alpha)P2(i) \tag{2.10}$$

$$C2(i) = (1 - \alpha)P1(i) + \alpha P2(i) \tag{2.11}$$

où α est un coefficient de pondération aléatoire adapté au domaine d'extension des gènes (il n'est pas nécessairement compris entre 0 et 1, il peut prendre des valeurs dans l'intervalle [-0.5, 1.5], ce qui permet d'engendrer des points entre, ou à l'extérieur des deux gènes considérés).

2.4.2.5 Opérateur de mutation

Pour les problèmes discrets, la mutation consiste, généralement, à tirer aléatoirement un gène dans le chromosome et le remplacer par une valeur aléatoire dans un

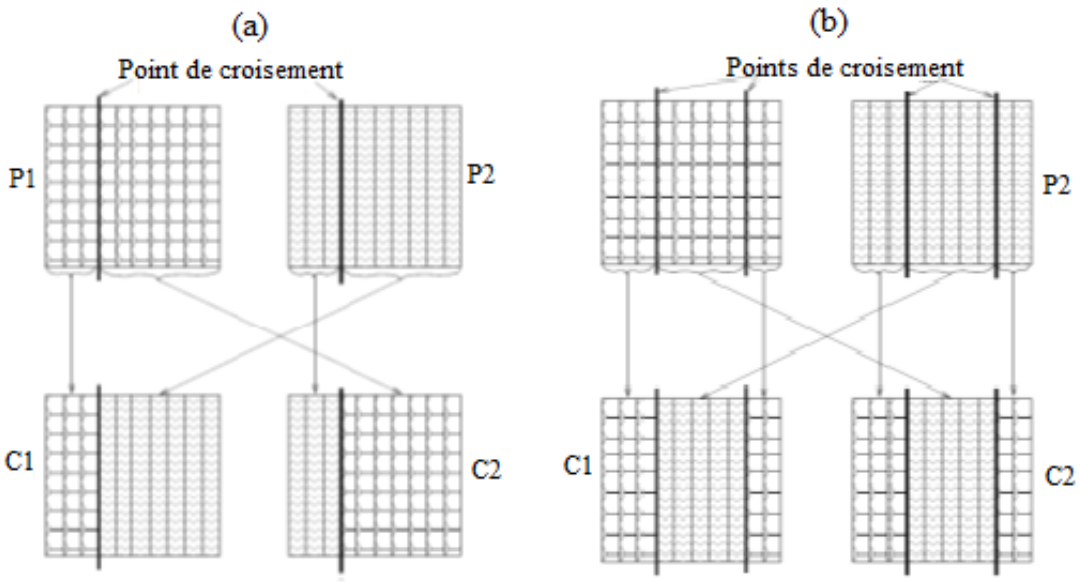

Figure 2.7: Croisement (a) à un point ; (b) à deux points [56]

intervalle défini pour un codage réel (figure 2.8). Et pour un codage binaire elle consiste à changer la valeur du bit de gène de 0 à 1 ou inversement. Dans les problèmes continus, un bruit généralement gaussien est ajouté au gène tiré. L'écart-type de ce bruit est difficile à choisir a priori. L'opérateur de mutation apporte aux AGs la propriété d'ergodicité de parcours de l'espace. Cette propriété indique que l'algorithme génétique sera susceptible d'atteindre tous les points de l'espace d'état, sans les parcourir tous dans le processus de résolution. La mutation empêche aussi l'AG de tomber dans des extrêmes (minimas ou maximas) locaux. Cependant, la mutation ne devrait pas se produire très souvent, puisque l'AG va devenir aléatoire et peut être incontrôlable.

2.4.2.6 Principes de sélection

La sélection est la première étape appliquée aux individus d'une population dans chaque itération. Elle évalue la performance de chaque individu à l'aide de la fonction fitness pour éliminer les mauvais et reproduire des nouveaux survivants après croisement et mutation.

La forme et le type de cette fonction varient d'une application à une autre, et si on essaie de maximiser ou de minimiser le problème à résoudre. Ce qui rend le domaine d'application des AGs plus vaste. Dans la littérature, nous trouvons plusieurs principes de sélection plus ou moins adaptés aux problèmes qu'ils traitent. Parmi eux, nous citons :

- la sélection par loterie biaisée : cette méthode se base sur le principe de tirage aléatoire utilisé dans les roulettes de casinos avec une structure linéaire. Elle

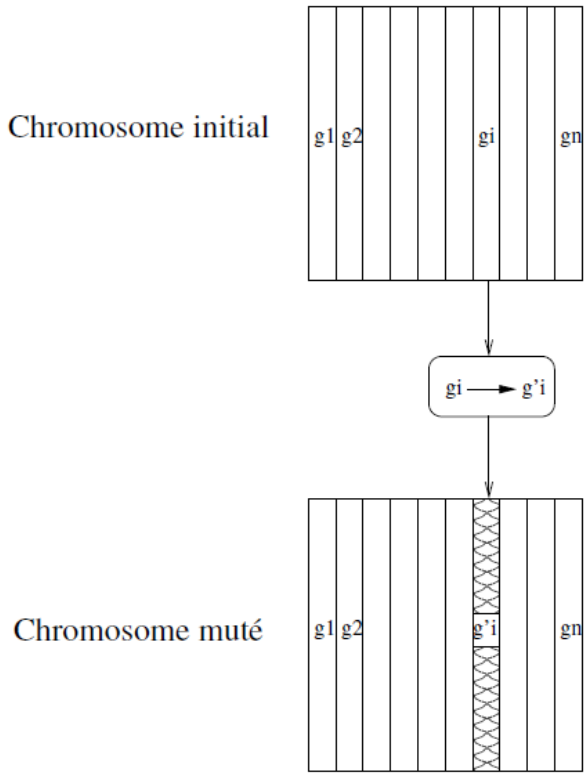

Figure 2.8: Principe de mutation [56]

consiste à associer à chaque individu un segment de longueur proportionnelle à son fitness. Puis, ces segments sont concaténés sur un axe normalisé (varie de 0 à 1). Ensuite, on tire un nombre aléatoire réel de cet axe et on regarde à quel segment il appartient. Avec ce système, les grands segments, c'est-à-dire les bons individus, seront plus souvent choisis que les petits ;

- la sélection élitiste ou par rang : dans cette méthode les individus, initiaux ou générés par croisement et mutation, sont classés par rapport aux leurs fitness. Ensuite les k meilleurs seront sélectionnés, tel que k est la taille de la population;
- la sélection par tournoi : cette méthode fonctionne en prenant aléatoirement un ensemble d'individus, puis on sélectionne ceux avec la meilleure fitness pour chaque ensemble. Ce tournoi se répète jusqu'on arrive à un nombre de tournois précisé ou à k individus sélectionnés, tel que k est la taille de la population.

2.5 Conclusion

Dans ce chapitre nous avons familiarisé avec la méthode de synthèse par sélection d'unité. Nous avons présenté son principe et tous ces éléments de base nécessaire pour avoir un système de synthèse par sélection d'unités. Nous avons aussi vu quelques outils informatiques qui peuvent être utile pour l'amélioration d'un tel

système. Ces informations sont des préparations pour l'élaboration de notre système de récitation automatique du Saint Coran, qui sera discuté dans le chapitre suivant.

CHAPITRE 3 : ELABORATION DU SYSTÈME HQ_TTS

3.1 Introduction

Notre objectif dans cette thèse consiste à développer un système capable de lire automatiquement et de façon correcte les versets du Saint Coran (SC). Ce but peut être réalisé à l'aide d'un système de synthèse à partir du texte. Et pour arriver à la meilleure qualité vocale, la Synthèse par Sélection d'Unité (SSU) offre le bon choix d'après les chapitres précédents. Ce chapitre décrit les étapes suivies dans le développement d'un tel système que nous le notons « HQ_TTS : Holy Quran Text-To-Speech ». Elles consistent d'abord, de l'élaboration d'une grande Base de Données (BD) riches d'unités. Ensuite, le texte entré se convertit en une représentation phonétique de sa prononciation. Après, les meilleures unités vont être sélectionnées de cette BD, puis concaténées.

3.2 Initiation sur le système HQ_TTS

Le HQ_TTS peut être considéré comme un système de synthèse en Arabe Standard (AS), puisque le Saint Coran (SC) s'écrit et se lit en Arabe. La seule déférence est que sa lecture (récitation) nécessite d'autres règles que celles existent déjà dans l'AS, appelées règles de *tajweed*. Avec ces dernières, de nouveaux phonèmes s'apparaissent (comme les voyelles à double ou triple durée « le *madd* »), et d'autres phénomènes phonologiques qui se présentent ou se bien réalisent comme l'assimilation (dans le cas où le [n] est suivi par les phonèmes [j], [r], [m], [l] ou [n]), l'emphase autour les phonèmes [s̪], [x], [d̪], [t̪], [z̪] ou [q], etc. La définition pratique de *tajweed* dite qu'elle est une science dédiée pour perfectionner la récitation du SC afin de la perfectionner, tout en prononçant correctement chaque phonème sans exagération et avec douceur. La sainteté du Coran interdit d'utiliser les règles de *tajweed* pour lire des textes arabes normaux [59].

3.3 Construction de la Base de Données (BD)

La base de données est un module très important dans les systèmes SSU. Elle doit être riche d'unités avec différentes caractéristiques prosodiques et contextuelles pour arriver aux meilleures performances. Dans la construction du système HQ_TTS l'élaboration d'une telle structure a été comme suit.

3.3.1 Préparation du corpus

Le corpus préparé pour le HQ_TTS est issu des enregistrements téléchargés d'un site web [60]. Parmi les récitations qui existent dans ce site, nous avons choisi les enregistrements faites par Dr « Aymen Rouchdi Souaid » selon le type de récitation (*riwayah*) « Hafs An Aasim ». Cette dernière est la plus répandue dans le monde islamique. De plus, les enregistrements utilisés sont, originalement, destinés à l'enseignement des règles de *tajweed*. En d'autres termes, ils sont favorisés à cause de leur lecture normale et monotone du Coran. Cependant, la majorité des récitations destinées à l'écoute inclue de la mélodie et variations toniques. Dans cette préparation, nous avons étudié puis déterminé les combinaisons des diphones qui existent dans le SC avec leurs entourages contextuels (car il y a des combinaisons qu'ils n'existent même pas dans l'AS comme : ص - ج ([sˤ- ʤ]). Visant à collecter au minimum toutes ces combinaisons possibles dans deux contextes différents au moins, nous avons opté pour 726 versets et parties de verset, de longueur d'un à dix mots. Ces fichiers audios ont été pris de résolutions 16 bits / 44.1 kHz et 16 bits / 22.05 kHz. Ils arrivent à une durée totale de 45 mn (environ 3.85 % de la durée d'enregistrement totale du SC).

3.3.2 Segmentation et annotation

Après la préparation du corpus, les versets choisis sont analysés et segmentés en petites unités (diphones et polyphones), pour extraire leurs fichiers audio correspondants et les mettre dans la BD. À l'aide du logiciel « Praat », cette étape a été effectuée manuellement, où nous avons basé sur notre écoute, l'audiogramme et le spectrogramme de ces enregistrements. Pour faciliter cette segmentation, les bords (frontières) de chaque phonème sont d'abord définies. Ensuite, avec l'écoute et un peu d'ajustement de ces frontières les diphones et polyphones sont définis (figure 3.1).

Dans le système HQ_TTS, le diphone a été pris comme unité de base pour ne pas, trop, encombrer notre BD par un choix d'unité plus grand. De plus, ça a été pour éviter le maximum de discontinuités spectrales lors de la concaténation des unités plus petites « comme le phonème ». En général, au cours de la segmentation, les frontières des diphones ont été positionnées au milieu des voyelles et les consonnes fricatives (figure 3.2 a). Tandis que l'emplacement du bord pour les occlusifs a été fait après l'explosion (figure 3.2 b).

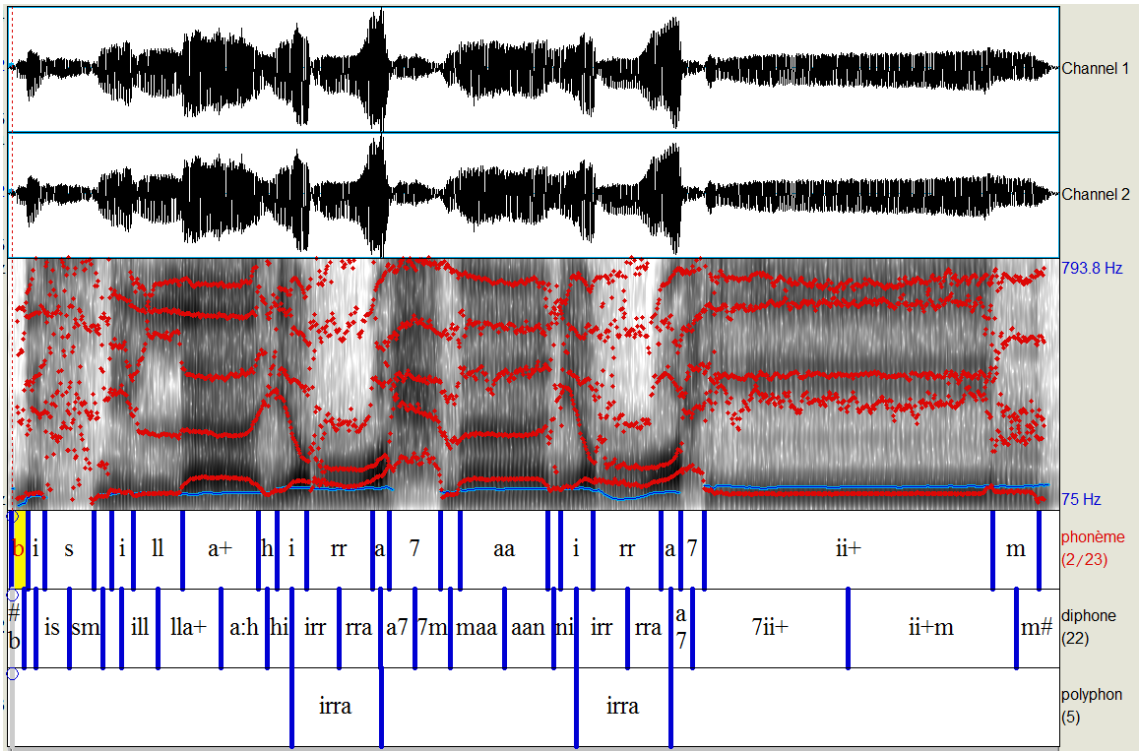

Figure 3.1: Segmentation de la phrase « [bismi llahi rraħmaani rraħiim] »

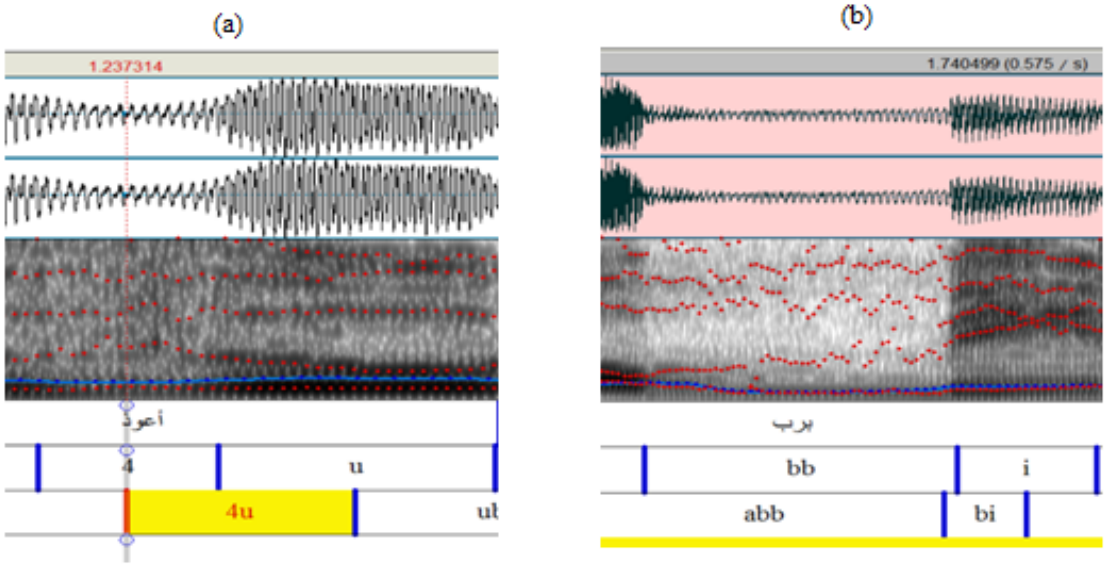

Figure 3.2: Exemple de segmentation d'une : (a) consonne fricative et une voyelle, (b) consonne occlusive

En plus des diphones, notre BD comprend aussi des polyphones (une constitution de trois à quatre phonèmes). Nous avons adopté cette unité pour couvrir les phonèmes à très courte durée comme la *hamza* «أ» [ʔ], ou les phonèmes qui sont difficiles à segmenter comme les semi-voyelles (figure 3.3).

Après l'extraction des segments sonores définis, nous avons les nommé par des codes alphanumériques pour les phonèmes constituants l'unité et certaines de leurs propriétés contextuelles. Comme la figure 3.4 montre, ce nom est composé de quatre parties :

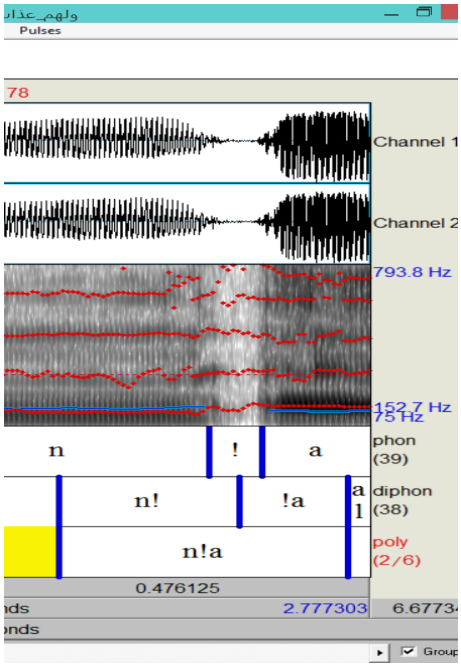

Figure 3.3: Exemple de segmentation de la *hamza* [ʔ]

- (1): les phonèmes constituant l'unité, représentés par un code inspiré de l'API « Alphabet Phonétique International », et se base sur le principe que chaque phonème arabe est représenté par un seul caractère du clavier (tableau 3.1) [61];
- (2): deux chiffres représentent la position du mot dans la phrase et la position de l'unité dans le mot. Ils prennent les valeurs {1, 2 ou 3} pour indiquer respectivement une position initiale, médiane ou finale ;
- (3) et (4): codes numériques indiquent respectivement les phonèmes gauche et droit de l'unité. Ces numéros sont affectés aux phonèmes arabes selon leur ordre présenté dans le tableau 3.1, y a compris le silence [#] codé par 35.

  Cette étape de segmentation et annotation abouti à 11112 unités.

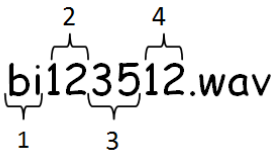

Figure 3.4: Annotation des segments sonores extraits

Ensuite, chaque segment sonore a été analysé une autre fois à l'aide du logiciel « speech analyzer » comme illustre la figure 3.5. Cette étape a été faite pour extraire de nouveaux fichiers contient certaines caractéristiques acoustiques des unités (les valeurs de l'énergie, la fréquence fondamentale (F_0), la durée, les formants, etc.) comme présente la figure 3.6. Ils ont été aussi enregistrés dans la BD avec le même nom que les fichiers audio.

Tableau 3.1: Code d'Alphabet Phonétique International (API) et code proposé des caractères arabes

Caractère en AS	API	Code proposé	Caractère en AS	API	Code proposé
أ	[ʔ]	[!]	ع	[ʕ]	[3]
ب	[b]	[b]	غ	[ɣ]	[g]
ت	[t]	[t]	ف	[f]	[f]
ث	[θ]	[8]	ق	[q]	[q]
ج	[ʤ]	[5]	ك	[k]	[k]
ح	[ħ]	[7]	ل	[l]	[l]
خ	[x]	[x]	م	[m]	[m]
د	[d]	[d]	ن	[n]	[n]
ذ	[ð]	[4]	هـ	[h]	[h]
ر	[r]	[r]	و	[w]	[w]
ز	[z]	[z]	ي	[y]	[j]
س	[s]	[s]	اَ	[a]	[a]
ش	[ʃ]	[c]	اَا	[aa]	[aa]
ص	[ṣ]	[$]	اُ	[u]	[u]
ض	[ḍ]	[£]	اُو	[uu]	[uu]
ط	[ṭ]	[6]	اِ	[i]	[i]
ظ	[ẓ]	[%]	اِي	[ii]	[ii]

En plus de toutes ces données, la BD comprend aussi les 12 valeurs MFCC de chaque segment sonore, qui ont été calculées à l'aide du toolbox « HTK MFCC MATLAB » [62].

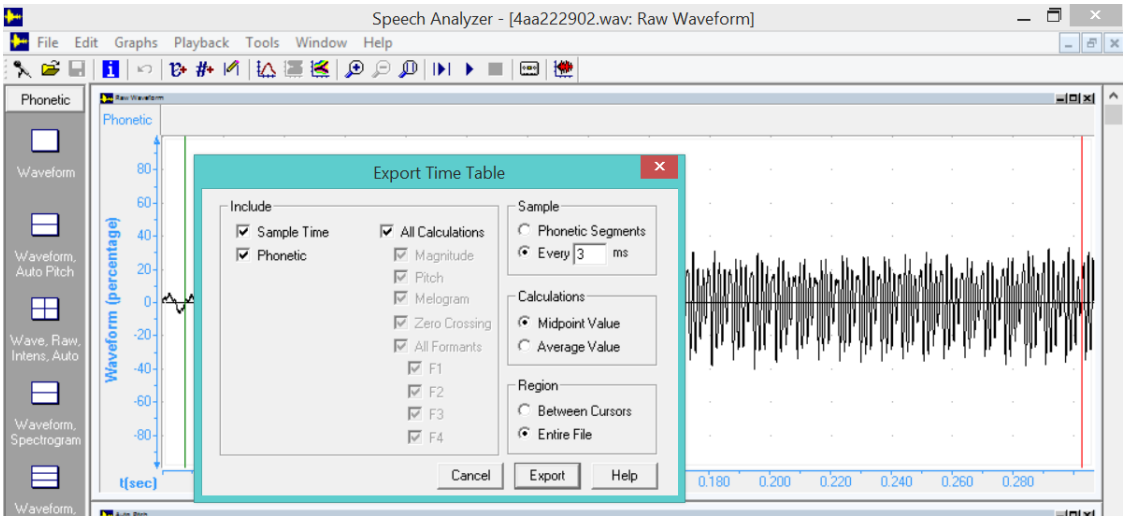

Figure 3.5: Analyse par « Speech analyzer » et l'extraction des fichiers des caractéristiques acoustiques de type «.sft »

4aa222902.sft - Bloc-notes

Fichier Edition Format Affichage ?

```
\table time every 3ms
Time    Int(dB) Pitch(Hz)       RawPitch        SmPitch Melogram(st)    ZCross  F1(Hz)  F2(Hz)  F3(Hz)  F4(Hz)
0.000   -34.8                                   1
0.003   -28.8   120.4                           2
0.006   -28.8   121.9           121.8   46.60   1
0.009   -24.6   123.1           122.7   46.74   1
0.012   -22.0   124.4           123.9   46.98   2
0.015   -22.0   125.9           125.5   47.37   2
0.018   -20.7   126.4           126.1   47.48   1
0.021   -19.9   126.8           126.6   47.53   2       520.0   1427.1  2349.6  3514.1
0.024   -19.9   128.3   130.2   128.3   47.64   1       520.0   1427.1  2349.6  3514.1
0.027   -19.0   128.8   130.2   128.8   47.73   2       520.0   1427.1  2349.6  3514.1
0.030   -18.1   129.1           129.1   47.82   2       520.0   1427.1  2349.6  3514.1
0.033   -18.1   130.6   130.6   130.6   48.03   1       520.0   1427.1  2349.6  3514.1
0.036   -17.1   131.8   132.3   131.8   48.20   2       520.0   1427.1  2349.6  3514.1
0.039   -16.1   132.8   134.8   132.8   48.34   2       443.6   1373.9  2437.2  3626.3
0.042   -16.1   134.4   132.3   134.4   48.48   1       443.6   1373.9  2437.2  3626.3
0.045   -15.0   135.5   134.9   135.5   48.59   1       443.6   1373.9  2437.2  3626.3
```

Figure 3.6: Partie d'un fichier « sft » contenant quelques caractéristiques acoustiques de l'unité «4aa222902 »

3.4 Elaboration du HQ_TTS

Comme tout système de synthèse à partir du texte le HQ_TTS se compose principalement de deux parties : le Traitement du Langage Naturel (TLN) et le traitement automatique de la parole (sélection + concaténation) comme présente la figure 3.7.

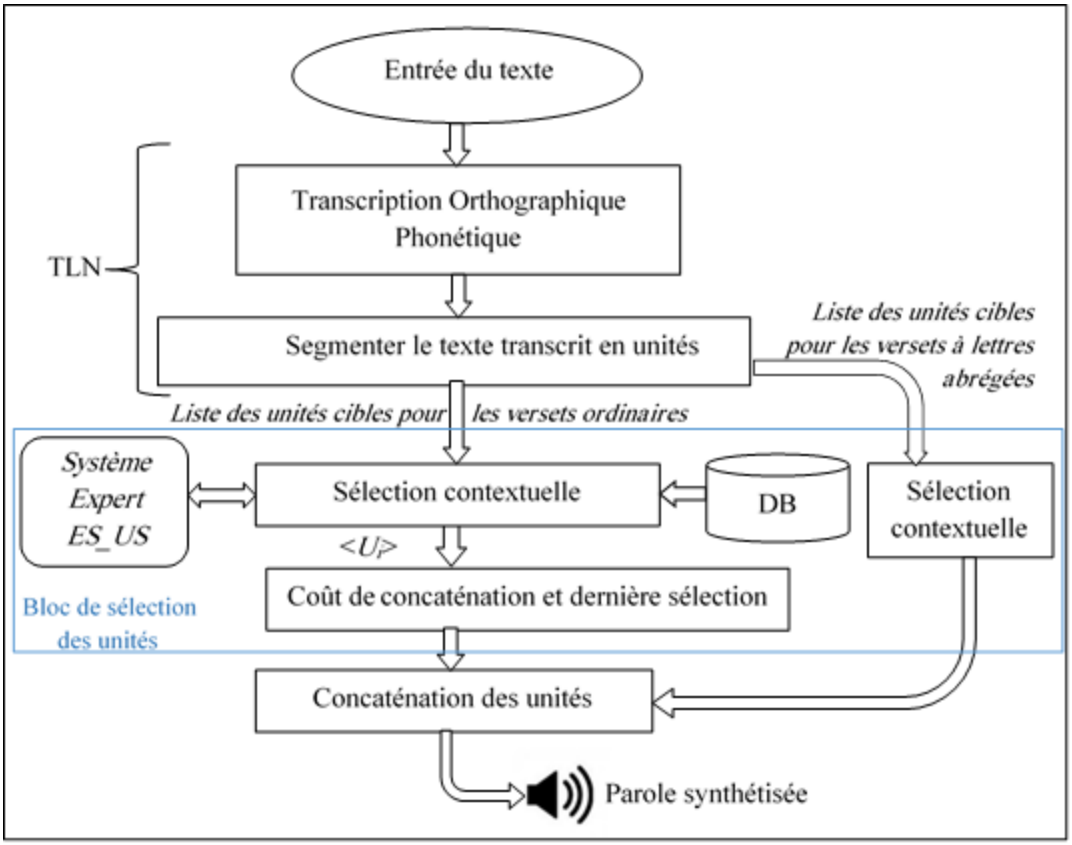

Figure 3.7: Schéma synoptique du système HQ_TTS

3.4.1 Traitement du Langage Naturel (TLN)

Ce module consiste à traiter le texte entré, et le convertir en une écriture phonétique correspondante à sa prononciation. Son but principal est d'arriver à une liste des unités cibles qui s'accorde au texte à synthétiser.

3.4.1.1 Transcription Orthographique Phonétique (TOP)

La TOP désigne la conversion du texte écrit en une représentation symbolique de sa prononciation. Dans la littérature, nous trouvons principalement trois méthodes de transcription, où le choix de la plus efficace dépend de la langue à synthétiser et la complexité de son système d'épellation [63]:

- la transcription par dictionnaire : se base sur la conversion de chaque mot ou article du texte à partir d'un dictionnaire contient l'écriture phonétique (prononciation) de tous les mots ou morphèmes du langage. C'est une méthode qui nécessite une très grande mémoire de stockage, mais elle est efficace pour les langues ayant beaucoup d'exception de prononciation dans leurs mots ;
- la transcription par règles : se fait par l'application des règles régulières et bien définies. Elle s'applique aux langues ayant un système d'épellation régulier, et un taux de conversion direct graphème - phonème élevé ;
- la transcription par traitement des données : qui se base sur les techniques d'apprentissage automatique des machines pour apprendre puis transcrire correctement les mots du langage.

En général, le système HQ_TTS suit la méthode de transcription par règles, l'Arabe peut se considérer comme une langue avec un système d'épellation régulier quand elle est écrite avec tous ses signes diacritiques (comme le cas du SC). Pour un meilleur résultat, un dictionnaire d'exception a été intégré dans cette phase pour quelques mots spéciaux (qui ne subissent à aucune règle implémentée). Pour faciliter l'implémentation et la modification des règles de conversion, la TOP a été subdivisée en plusieurs modules comme suit (figure 3.8) [64].

3.4.1.1.1 Prétraitement du texte

Comme présente la figure 3.8 la transcription commence par une phase de prétraitement de texte qui traite les abréviations, les ponctuations, caractères spéciaux, chiffres, etc. Puisque l'entrée du système HQ_TTS est un texte coranique simple, donc

nous ne trouvons ni des abréviations ni des chiffres. De plus, le texte est saisi par le clavier, donc le peu des caractères spéciaux qui paragraphes (le point et le point-virgule respectivement). Dans cette phase, le existent dans le SC ne peuvent pas être introduits. Dans ce cas-là, notre système traite que quelques ponctuations indiquant la fin des phrases ou des HQ_TTS fait aussi la distinction entre les versets ordinaires et celle à lettres abrégées [1] qui ont un mode de lecture spécial. Ce dernier type des versets se détecte par la non existence des voyelles dans le texte.

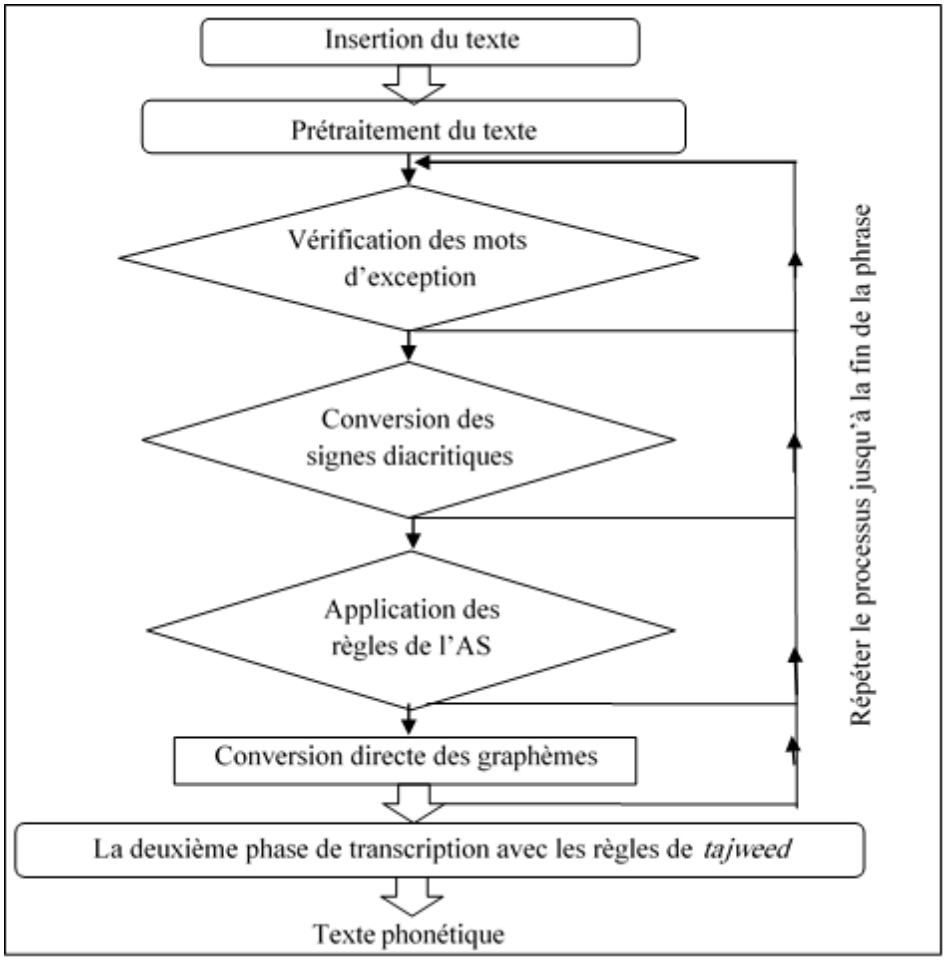

Figure 3.8: Etapes de la Transcription Orthographique Phonétique (TOP) dans le système HQ_TTS

3.4.1.1.2 Analyse du texte alphabétique et la première phase de transcription

Cette phase consiste à transférer le texte prétraité à une écriture phonétique selon notre code proposé (tableau 3.1). Dans le cas des versets à lettres abrégées la transcription se fait par une conversion directe des lettres (consonnes) en leurs codes correspondants. Tandis que, pour le reste des versets les étapes suivantes doivent être vérifiées et traitées par ordre pour chaque caractère du texte.

[1] Ils se présentent dans le début de 29 sourats et se composent par une combinaison d'une à cinq lettres parmi la liste {ي ، هـ ، ن ، م ، ل ، ك ، ق ، ع ، ط ، ص ، س ، ر ، ح ، أ}, comme :"ألر".

- **traitement des mots d'exception :** lorsque le texte coranique est entré dans l'espace du travail de MATLAB, il va automatiquement s'interpréter comme un code numérique (en Windows-1256). Puis il va être analysé lettre par lettre et mot par mot pour déterminer s'il contient l'un des mots d'exception (des mots avec une prononciation spéciale, et leur transcription ne subit à aucune règle de lecture arabe). Ensuite, et par comparaison du code numérique du mot avec ceux existant dans le dictionnaire des exceptions, la transcription s'effectue directement à partir de ce dernier. Ce dictionnaire a été construit basant sur les mots listés dans les travaux de [63, 65, 66], et après une analyse des mots du SC et leur prononciation. Le tableau 3.2 présente quelques mots de ce dictionnaire;

Tableau 3.2: Transcription phonétique de quelques mots d'exception dans le dictionnaire

Mot spécial	Transcription phonétique
هَؤُلَاءِ	[haa!ulaa!]
أُولَئِك	[!uulaa!ik]
هَذَا	[haa4aa]

- **conversion des signes diacritiques :** après avoir traité les mots d'exception, l'algorithme de la TOP teste si le caractère analysé, actuellement, correspond au l'un des signes diacritiques pour le convertir comme suit :
  - les harakaat se représentent comme des voyelles courtes (اَ -> [a], اُ -> [u], اِ -> [i]) ;
  - le *tanween* se transcrit en une voyelle + [n], donc (اً -> [an], اٌ -> [un], اٍ -> [in]) ;
  - la chadda est transcrite comme un doublement de consonnes (نّ -> [nn]) ;
  - le *madd* est transcrit comme une double voyelle (اَا -> [aa],اُو -> [uu], اِي -> [ii], آ -> [!aa]) ;
  - symboliser tous les types de la *hamza* avec un seul caractère phonétique (أ، ء، ؤ ، ئ -> [!]).
- **application des règles principales de la lecture en AS :** lors de la conversion des signes diacritiques, la transcription des versets entrés se continue en appliquant les règles de la lecture arabe. Cette étape consiste à remplacer chaque code numérique (du Windows-1256) par aucun, un ou deux caractères phonétiques, tout dépend de la lettre à transcrire, sa position et ses caractères voisins à gauche

et à droite. Puisque le SC est un document en Arabe complètement diacritisé, les règles à appliquer peuvent être limitées aux ;

- *règle de «ال» de définition :*
  - au début de la phrase : le «ا» se transcrit en [!a], et le «ل » a deux cas. Elle se transcrit en [l] si elle est suivie d'une lettre lunaire {أ، ب، ج، ح، خ، ع، غ، ف، ق، ك، م، هـ، و، ي}.Ou bien, Elle se supprime si elle est suivie par une lettre solaire {ت، ث، د، ذ، ر، ز، س، ش، ص، ض، ط، ظ، ل، ن}, avec la gémination de cette dernière (doublant son caractère phonétique).
  - au milieu de la phrase, le «ا» se supprime, et la même règle précédente s'applique au «ل». Par exemple : «الْقَارِعَةُ» -> [!alqaariratu] ; «و الشَّمسِ» -> [waccamsi].
- *règle de la hamzat-wasl :* elle s'agit du caractère «ا» quand il est silencieux au début d'un mot. Tout dépend son contexte la *hamza* est transcrite comme suit :
  - au milieu de la phrase, elle se supprime ;
  - au début de la phrase, elle se transcrit en [!i] ou [!u], tout dépend le type du mot qui la contient (verbe ou nom). Afin de simplifier cette étape sans avoir besoin d'étudier le type du mot, la transcription se fait par l'application de quelques règles avec un dictionnaire d'exception : Si la troisième lettre du mot est suivie par «أ» ou bien le mot fait partie de la liste {ابْن ، ابْنَت ،امْرُؤ ، امْرَأَت امْرَأَة ، اثْنَتَ ، اثْنَ ، اسْم}, la *hamza* est transcrite en [!i]. Sinon, si la troisième lettre est suivie par «ُأ», la *hamza* est transcrite en [!u] sauf pour les mots {امْضُو ، اقْضُو ، امْشُو ، ابْنُو ، ائْتُو، ايْتُو} où elle se transcrit en [!i].
    Exemple :« اِقْـرَأْ » -> [!iqra!] ; « اِمْـضُـو » -> [!im£uu].
- *règle du alif-maqsûra :* le « ى » est un type du *alif* (ا) silencieux qui se trouve à la fin de quelques mots. Dans le système HQ_TTS deux règles sont appliquées :
  - si elle est précédée par le signe « اَ » elle disparaît et la voyelle [a] devient [aa] ;
  - sinon, si elle est précédée par le *tanween* « اً » elle disparaît sans aucun autre changement.

Exemple : «عَلَـى» -> 3alaa] ; «هُدًى» -> [hudan] .

- **conversion directe des graphèmes :** après le passage par toutes les étapes précédentes, le reste des caractères est transcrit par une conversion graphème-phonème directe comme indique le tableau 3.1. De plus, le signe du (*sukune*) « ٱ » est supprimé et le caractère [#] est ajouté au début et à la fin du texte transcrit pour indiquer le silence.

3.4.1.1.3 Application des règles de *tajweed*

Suivant le même principe comme dans (3.4.1.1.2), les règles de *tajweed* doivent être aussi vérifiées et s'appliquées dans un ordre spécifique, et alors, une transcription finale du texte sera réalisée. Afin d'organiser cette étape et réduire le temps d'exécution, ces règles ont été regroupées en fonction du type et position du phonème dans le mot. Car certaines règles fonctionnent à la fin du mot, d'autres s'appliquent à un type spécifique de consonnes (comme l'assimilation « *idgham* » avec le phonème [n]), etc.

- **règles à la fin de la phrase :** ce type de règles s'applique lorsque certains phonèmes figurent à la fin de la phrase comme :
  - la voyelle à la fin de la phrase se supprime, mais si elle est précédée par [t] (originalement « ة »), les deux sont supprimées et remplacées par le phonème [h]. Exemple : [#!alqaari3atu#] -> [# ! a l q aa r i 3 a h #] ;
  - les deux cas de *tanween* « ٱً ، إٍ » ([in] et [un]) sont supprimé. Cependant pour différencier entre les cas de vrais *tanween* et les cas qui se terminent par un véritable [in] ou [un], nous avons créé une liste des mots d'exception pour ce dernier. Exemple : [3aamilatun naa$ibatun#] -> [3 aa m i l a t u n-n aa $ i b a h#].

- **règles pour les phonèmes doubles :** lorsqu'un double phonème se présente, deux cas doivent être vérifiés :
  - s'il s'agit des phonèmes {q, 6, b, 5 ou d}, la règle de *qalqalah* est appliquée en remplaçant le phonème par le symbole de *qalqalah* présenté dans le tableau 3.3. Exemple : [!idfa3#] -> [!id_dfa3#] ;
  - s'il s'agit d'une voyelle, les règles de *madd* sont appliquées et la voyelle longue sera remplacée par le caractère du *madd* correspondant au cas trouvé

(tableau 3.3). Exemple : [jaa !ajjuhaa] -> [j aa+ ! a jj u h aa].

- règles des cas généraux : Ils regroupent le reste des règles de *tajweed* comme :
  - l'assimilation : lorsque le phonème [n] suivi par {j, r, m, w, l ou n}, une nouvelle représentation remplace ces deux phonèmes (tableau 3.3). Exemple : [!an ra!aahu#] -> [! a n-r a ! aa h#] ;
  - le *ikhfaa* : lorsque le phonème [n] est suivi par {$, 4, 8, k, 5, c, q, s, d, 6, z, f, t, £ ou %}, il sera changé par [n~] (tableau 3.3). Exemple : [waman kaana] -> [w a m a n∼ k aa n a].

Tableau 3.3: Caractères et codes utilisés pour les règles de *tajweed*

Caractère ou code	**Signification**
[aa+]	Représente le *madd* à quatre mouvements pour la voyelle[aa]
[aaa]	Représente le *madd* à six mouvements pour la voyelle [aa]
[~]	Rejoint le phonème [n] dans le cas du *ikhfaa* ([nc]=>[n~c])
[-]	Se met entre deux phonèmes assimilés ([n-l])
[_]	S'ajoute dans le cas de la *qalqalah* ([bt]=>[b_bt])
[']	S'ajoute au phonème [m] qui a été [n] dans le cas du *iqlab* ([nb]>[m'b])

Tout en vérifiant les cas précédents et en appliquant les règles du *tajweed*, les espaces entre les mots sont supprimés à cause de la caractéristique continue de la parole.

3.4.1.2 Analyse du texte et segmentation en unités

Après la transcription, le texte résultant est analysé pour définir une liste des unités cibles. Pour le type ordinaire des versets, le texte est décomposé en diphones et polyphones. Tandis que pour les versets à lettres abrégées, cette étape se fait par leur décomposition en lettres qui les contiennent (phonétiquement en mots). Ensuite, chaque unité définie est accompagnée par ses caractéristiques contextuelles comme la notation des segments sonores dans la BD.

La figure 3.9 montre les étapes de transcription et segmentation du 2ème verset de *sourat* El-Falaq. L'une de ses unités cibles est « #m110034 ». Elle est composée du diphone [#m] situant dans le début du premier mot de la phrase (code de position = « 11 ». « 00 » indique que cette unité n'a pas de phonème précédent. « 34 » est un code pour le [i] qui est le phonème suivant.

De même, la figure 3.10 présente les étapes de transcription et segmentation du 1ier verset de *sourat* El-Baqara. « l2 01 24 » est l'une des unités cibles résultantes. Elle

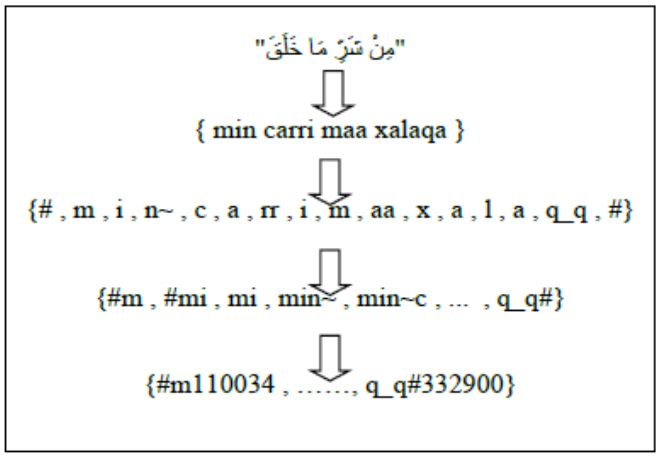

Figure 3.9: Transcription et segmentation d'un verset ordinaire »

consiste en la lettre [l] située dans la position médiane du verset, « 01 » et « 24 » sont les codes des lettres respectives précédente et suivante : [!] et [m].

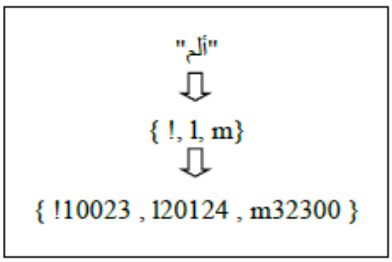

Figure 3.10: Transcription et segmentation d'un verset à lettres abrégées

3.4.2 Sélection des unités

Dans le système HQ_TTS, le processus de sélection des unités dépend de type du verset. Pour le cas ordinaire des versets, le système HQ_TTS commence par une recherche, dans la BD, des unités qui se composent des mêmes phonèmes que les unités cibles définies dans l'étape précédente. Ensuite, elles vont être structurées comme des listes d'unités candidates pour chaque cible, et les meilleures seront sélectionnées à partir de ces listes. Afin d'optimiser le temps de synthèse, notre algorithme de sélection a été divisé en deux étapes, contrairement à l'algorithme de base expliqué dans la section 2.1. Dans ce cas-là, les unités sélectionnées lors de la première étape « dite sélection contextuelle » vont être les nouvelles unités candidates pour la deuxième.

3.4.2.1 Sélection par fonction cible (sélection contextuelle)

Cette première phase tend à réduire le nombre des unités candidats par une sélection qui base sur leur ressemblance aux celles cibles. Le dégrée de correspondance entre une unité cible t_i et l'une de ses candidates u_i^j est évalué par un score cible TS_i^j.

Comme indique l'équation (3.1) ce dernier est obtenu par la somme des scores de comparaison de quatre caractéristiques contextuelles : le contexte droit $S_r(t_i, u_i^j)$, le contexte gauche $S_l(t_i, u_i^j)$ la position de l'unité dans le mot $S_{pw}(t_i, u_i^j)$ et la position du mot dans la phrase $S_{ps}(t_i, u_i^j)$. Dans la synthèse par concaténation, les longues unités sont favorisées (les polyphones par rapport aux diphones). Pour cela, une valeur « d_i^j» qui reflète la longueur de l'unité est aussi comptée dans la somme comme suit.

$$TS_i^j = S_r(t_i, u_i^j) + S_l(t_i, u_i^j) + S_{pw}(t_i, u_i^j) + S_{ps}(t_i, u_i^j) + d_i^j \tag{3.1}$$

Après le calcul du score TS_i^j pour chaque unité candidate, celles avec les plus grandes valeurs de score ($< U_i >$) sont prises pour passer à la deuxième phase, comme suit.

$$< U_i >= \underset{j}{argmax}(TS_i^j) \tag{3.2}$$

Comme nous avons montré dans le chapitre précédent, l'efficacité de sélection dépend du nombre et type des caractéristiques utilisées ainsi que leur pondération. Pour cela, deux approches d'affectation des scores S_r,S_l, S_{pw} et S_{ps} ont été implémentées et testées.

3.4.2.1.1 Affectation des scores par les Algorithmes génétiques (AGs) et l'Algorithme Génétique actif interactif (AGai)

Les AGs et l'AGai sont l'une des approches utilisées dans la pondération des caractéristiques utilisées dans la fonction du score cible. Pour l'implémentation de ces techniques, nous avons basé sur les travaux d'Alias [31, 35, 67], où les scores Sx sont déterminés comme suit :

$$S_x = w_x \times A(t_i, u_i^j) \tag{3.3}$$

avec :

x désigne la caractéristique contextuelle (x= {l, r, pw, ps}) ;

S_x est le score de la caractéristique « x » ;

w_x est le poids de la caractéristique « x » ;

$A(t_i, u_i^j)$ est une fonction binaire qui indique le résultat de la comparaison entre l'unité cible et candidat. Elle reçoit la valeur « 1 » s'il y a une correspondance totale entre les deux unités et « 0 » dans le cas contraire.

Le but des fonctions l'AG et l'AGai dans ce processus est de trouver les poids optimaux w_x.

- Premièrement la pondération des caractéristiques par les AGs a été configurée pour chaque unité. En utilisant la fonction « ga » du MATLAB, cette optimisation a été faite comme suit :

  – initialisation : la population initiale P_i de cette optimisation a été fixée de manière empirique à sept éléments composés de quatre nombres réels, déterminés aléatoirement entre 0 et 5 reflétant les poids à optimiser w_x (équation 3.4) ;

  $$P_i = \begin{Bmatrix} w_l^1 & w_r^1 & w_{pw}^1 & w_{ps}^1 \\ \vdots & \vdots & \vdots & \vdots \\ w_l^7 & w_r^7 & w_{pw}^7 & w_{ps}^7 \end{Bmatrix} \tag{3.4}$$

  – calcule du fitness et sélection des bons éléments : pour chaque type d'unité à trouver ses poids optimaux, nous prenons un exemple comme unité cible et nous cherchons ses meilleures correspondantes suivant les équations (3.2) et (3.3). Puis, la fitness de chaque élément de la population est calculée comme la distance en 12 valeurs MFCC entre l'exemple et son unité correspondante. Après ce calcule, les éléments qui ont les minimums valeurs de fitness sont les plus probables d'être sélectionnés pour l'étape de reproduction (croisement et mutation) ;

  – application des opérations de croisement et mutation aux éléments sélectionnés en prenant les valeurs par défaut de la fonction « ga » ;

  – la condition d'arrêt de ce processus a été fixées par ordre à une valeur minimale de fitness (dépend de l'unité cible), un temp de calcul limite à 15 mn ou la condition par défaut de la fonction « ga » qui est la stabilisation de la valeur de fitness.

- Deuxièmement l'AGai est utilisé pour trouver la bonne pondération des caractéristiques d'unité pour une phrase donnée, comme suit (figure 3.11) :

  D'abord l'algorithme est initialisé avec un ensemble de configurations aléatoires des poids, constituant une population de 6 éléments avec des valeurs entre 0 et 5. Ensuite, cette population est utilisée pour synthétiser six versions différentes pour la phrase cible. Après, ces phrases synthétisées sont évaluées subjectivement (en termes de qualité de parole) dans un tournoi par paires en 2 rotations. Cela veut dire que chaque phrase est évaluée deux fois par un score allant de 1 « très mauvaise qualité » à 5 « très bonne qualité ».

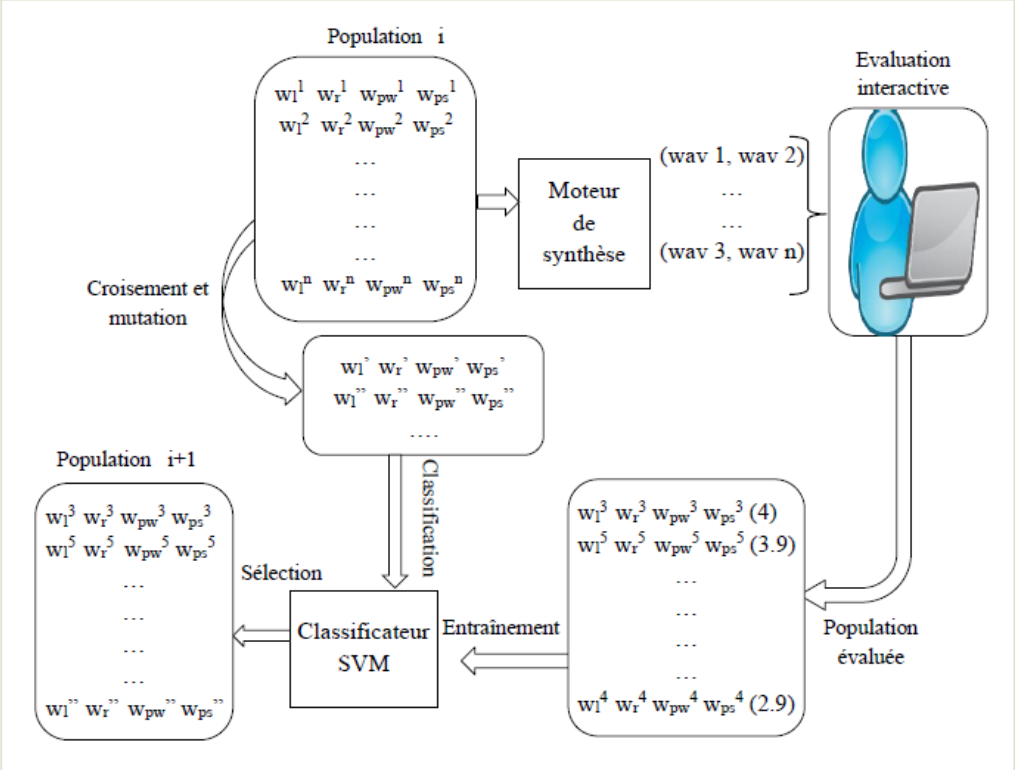

Figure 3.11: Schéma explicatif du processus d'entraînement des scores par AGai

Ces scores sont ensuite utilisés pour entraîner un classificateur « Support Vector Machine (SVM) » à deux classe bonne et mauvaise. Par un calcul de la dégradation D en valeurs MFCC pour chaque phrase (équation 3.5), elle est considérée comme bonne c'est son score est supérieur à la moyenne de tous les scores, et mauvaise dans le cas contraire. Ces phrases sont aussi ordonnées pour choisir les trois meilleurs passant aux opérations de croisement et de mutation. Ensuite, les éléments croisés ou mutés sont utilisés pour synthétiser de nouvelles phrases qui seront aussi classées en bonnes ou mauvaises. De même, ce classement se fait par le modèle SVM entraîné selon leur dégradation MFCC. À la fin, la nouvelle population sera constituée des meilleurs éléments de l'ancienne, et complétée par les bons éléments reproduits.

$$D = \sum_{i=1}^{12} \frac{(|x_i(1) - x_i(2)| + |x_i(2) - x_i(3)| + .. + |x_i(n-1) - x_i(n)|)}{n-1}) \qquad (3.5)$$

tel que : x est l'un des 12 vecteurs caractéristiques MFCC de taille n.

Ce processus se répète pour six itérations seulement, pour éviter la fatigue de l'auditeur. De plus, les meilleures configurations des poids déterminées par chaque auditeur sont utilisées comme des valeurs initiales du prochain processus par une autre personne. Une fois cet ajustement des poids est terminé, la dernière configuration pour chaque caractéristique est stockée dans la BD pour être utilisée, après, dans le calcul de la fonction du score cible.

3.4.2.1.2 Affectation des scores par les Systèmes Experts (SE)

L'AS et surtout celle du SC est caractérisée par ses phénomènes phonétiques et phonologiques (la gémination, l'emphase, l'assimilation, etc.), qui sont liés, principalement, au type des phonèmes et leur position, ce qui justifie notre choix des caractéristiques utilisé dans l'équation (3.1). Ces phénomènes, affectent beaucoup les caractéristiques acoustiques et prosodiques des unités adjacentes (la durée, l'énergie, la fréquence fondamentale, etc.). Comme par exemple, la figure 3.12 et le tableau 3.4, montrent l'influence de l'emphase sur la voyelle adjacente. Il est bien clair que la voyelle [a] prise à partir de deux contextes similaires (deux mots terminaux [falaq] et [xalaq]), a connu un abaissement des valeurs de F_2 quand elle est précédée par le phonème emphatique [x].

Tableau 3.4: Valeurs des formants et de la fréquence fondamentale d'une voyelle « a » prononcée par un homme, extraites à l'aide du software speech analyzer [69]

Temps (ms)	**La voyelle [a] précédée par la consonne emphatique [x]**					**La voyelle [a] précédée par la consonne [f]**				
	F_0(Hz)	F_1(Hz)	F_2(Hz)	F_3(Hz)	F_4(Hz)	F_0(Hz)	F_1(Hz)	F_2(Hz)	F_3(Hz)	F_4(Hz)
0.000						138.4				
0.015	165.3	529.9	**827.5**	2638.6	3701.6	144.4	569.5	**1356.1**	2557.1	3600.2
0.030	164.1	529.9	**827.5**	2638.6	3701.6	143.6	569.5	**1356.1**	2557.1	3600.2
0.045	161.8	549.9	**827.0**	2862.0	3796.1	143.1	578.0	**1425.2**	2626.6	3601.6
0.060	161.3	581.8	**822.6**	2945.2	3857.5	141.5	578.4	**1428.3**	2663.9	3656.8
0.075	161.1	618.0	**833.6**	2939.4	3758.9	139.8	586.0	**1462.1**	2658.2	3646.2
0.090	162.5	618.0	**833.6**	2939.4	3758.9	139.8	584.9	**1504.9**	2654.7	3663.3
0.105	163.8	624.8	**860.5**	2937.2	3728.8	139.3	584.9	**1504.9**	2654.7	3663.3
0.120	165.0	641.0	**967.0**	2861.1	3688.1	139.1	579.0	**1526.8**	2650.9	3657.3
0.135	166.3	660.2	1125.8	2818.5	3675.5	138.7	572.0	1537.0	2666.9	3748.1
0.150	166.4	665.4	1316.1	2774.0	3707.3	138.4	564.9	1604.2	2652.0	3772.5
0.165	166.1	665.4	1316.1	2774.0	3707.3	137.9	564.9	1604.2	2652.0	3772.5
0.180	165.4	657.9	1467.2	2664.0	3817.3		554.2	1651.8	2676.6	3851.7
0.195	165.1	608.2	1531.3	2668.2	3917.3					
0.210		608.2	1531.3	2668.2	3917.3					

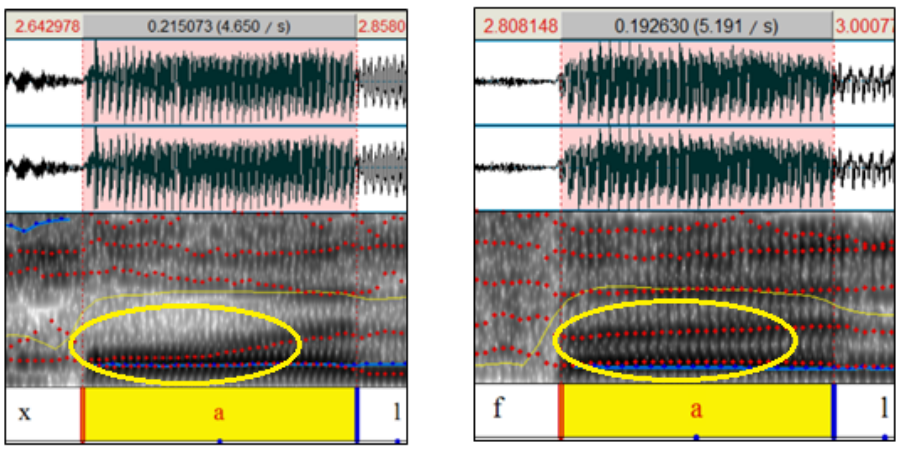

Figure 3.12: Spectrogramme et audiogramme de la voyelle « a » précédée par : un phonème emphatique « x » (à gauche), phonème non-emphatique « f » (à droite) [69]

Afin de bénéficier de ces caractéristiques dans la sélection, nous avons conçu un SE, noté « Expert System for Unit Selection (ES_US) » pour guider le processus d'affectation des scores. L'ES_US prend les caractéristiques des unités cible et candidate comme des entrées et déduit les scores, $S_r, S_l, S_p = \{S_{pw} + S_{ps}\}$ à attribuer pour ce cas. Il a été développé sous MATLAB, et il est composé des modules suivants (figure 3.13).

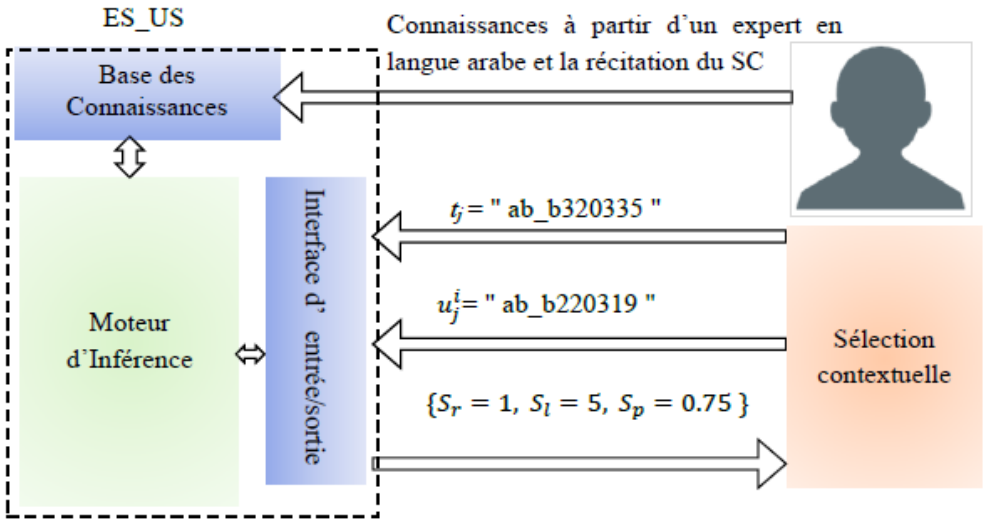

Figure 3.13: Schéma synoptique du système expert (ES_US)

- **l'interface d'entrée/sortie :** c'est l'outil d'interaction entre l'ES_US et le système de synthèse, lui-même. L'ES_US communique à travers cette interface avec l'algorithme de sélection, où il reçoit les nominations des unités cible et candidate comme des entrées et déduit les scores ($S_r, S_l ou S_p$) à attribuer à chaque caractéristique comme sortie;

- **la Base des Connaissances (BCc) :** elle regroupe les connaissances spécifiques au domaine d'utilisation du SE. La BCc est composée de :

  - la mémoire de travail, qui représente l'ensemble des faits décrivant l'état des entrées. Ils sont principalement classés en cinq catégories déduites des nominations des unités : longueur de l'unité ; type de l'unité à gauche ; type de l'unité à droite ; position de l'unité dans le mot ; et la position du mot dans la phrase, sous la forme : *catégorie* unité cible, unité candidate. Voici un exemple des faits définis pour le couple (t_i = ar121629, u_i^j= ara221624) :

    * *longueur de l'unité* {diphone, triphone} ;
    * *type de l'unité à gauche* {consonne emphatique, consonne emphatique} ;
    * *type de l'unité à droite* {voyelle, consonne} ;

  - *position de l'unité dans le mot* {première, deuxième} ;
  - *position du mot dans la phrase* {deuxième, deuxième}.

- la base des règles, qui est un ensemble de règles « si-alors ». Elle est construite à partir des caractéristiques de l'AS et les règles du *tajweed* du Coran. L'exemple suivant présente quelques règles liées au contexte gauche d'une consonne :
  - règle 1 : si l'unité candidate a une correspondance complète avec l'unité cible alors $S_l = 3$;
  - règle 2 : sinon si les unités cible et candidate sont précédées par des consonnes de différents types (par exemple fricative / occlusive) alors $S_l = 1$.

- **le Moteur d'Inférence (MI) :** il a été programmé pour émuler comment un réciteur expert du Coran prononce correctement chaque phonème, tout en analysant ses unités adjacentes et leur influence. Le MI reçoit ses données initiales de l'interface, puis il analyse le type (consonne emphatique, consonne assimilée, voyelle, etc.) et le contexte (type et caractéristiques des phonèmes voisins) des unités cible et candidate. Cette analyse ne se limite pas aux premiers voisins de l'unité, mais peut être étendue aux deuxièmes ou aux troisièmes dans certains cas. Après cela, les faits correspondant aux entrées actuelles sont définis et enregistrés dans la mémoire de travail. Ensuite, pour chacune des caractéristiques, les unités sont codées par un nombre de deux à quatre chiffres décrivant leurs contextes. À la fin, depuis ces codes, les règles appropriées seront tirées pour déduire directement les scores cherchés ($S_r, S_l et S_p$) (figure 3.14).

Les valeurs des scores dépendent du type d'unité, ainsi que du degré de ressemblance entre les contextes des unités candidate et cible. De plus, la valeur maximale du score n'est pas la même pour tous les cas. Elle varie en fonction de l'unité ou la caractéristique ciblée. Par exemple, la valeur maximale du score pour le contexte droit d'une consonne emphatique est «5», alors qu'elle est de «3» pour une consonne assimilée. Comme l'emphase est un phénomène spécifique à l'AS, connu par sa grande influence sur les caractéristiques acoustiques des phonèmes adjacents.

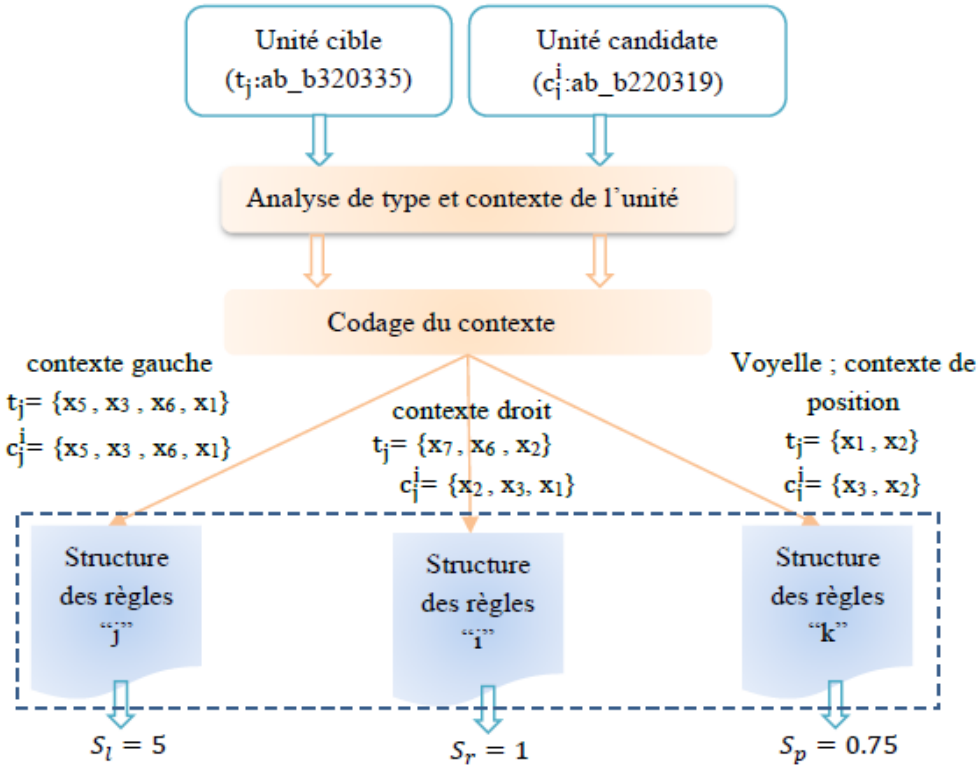

Figure 3.14: Exemple du processus d'affectation des scores par le moteur d'inférence

3.4.2.2 Sélection acoustique (finale)

Pour chaque deux unités consécutives u_{i-1} et u_i des unités restantes $<U_i>$, un coût de concaténation à cette jonction est calculé par la somme des trois sous-coûts pondérés. Reflétant la distorsion entre les unités u_{i-1} et u_i, ces sous-coûts sont calculés par la distance Euclidienne entre les valeurs de l'énergie : $E(u_{i-1}^m, u_i^j)$, de la fréquence fondamentale (F_0) : $F(u_{i-1}^m, u_i^j)$ et les 12 valeurs MFCC : $mfcc(u_{i-1}^m, u_i^j)$ au point de concaténation. De plus, une valeur de pénalisation f_i^j qui dépend du type de l'unité cible est ajoutée à la différence de F_0 en cas de contradiction (par exemple, au point de concaténation l'une des unités candidates a une valeur de F_0 et l'autre n'a pas de valeur). Les poids attribués aux sous-coûts ont été déterminés de façon empirique pour qu'ils soient de même ordre de grandeur ($w_f = 1$, $w_e = 2$ et $w_{mfcc} = 1/3$). Ensuite, à partir de ce calcul, la sélection des meilleures unités se fait par une recherche par programmation dynamique forward-backward comme suit [70].

3.4.2.2.1 Recherche Forward (RF)

Soit « n » le nombre de diphones composant la phrase à synthétiser. Commençant par les unités d'ordre 1 jusqu'à n, le coût de concaténation C_i^j à chaque étage i est calculé comme la somme des sous-coûts $E(u_{i-1}^m, u_i^j)$, $F(u_{i-1}^m, u_i^j)$ et $mfcc(u_{i-1}^m, u_i^j)$ avec le coût de la jonction précédente C_{i-1}^j. Après, les unités du côté droit de la jonction

sont considérées comme cibles et la meilleure candidate pour chacune d'entre elles est trouvée par minimisation des coûts calculés (équation 3.6). Ensuite, ces valeurs minimales (C_i^j) vont être gardées ainsi que les indices des unités trouvées (leur ordre dans l'étage) p_i^j (équation 3.7) et leur pas de saut (« 1 » pour un diphone, « 2 » pour un triphone, etc.). Les unités non choisies sont éliminées, ainsi que leurs coûts (C_i^j) et leurs chemins (du premier ordre jusqu'à p_i^j). À la fin, le meilleur chemin (la chaîne complète des meilleures unités « Chaîne Forward (CF) ») se déicide après le calcul des C_{n+1} puis un retour dans le réseau d'unités, suivant les indices et les pas de saut gardés (figure 3.15).

$$C_i^j = \begin{cases} 0 \quad si \quad i = 1 \\ min \begin{Bmatrix} C_{i-1}^1 & w_f F(u_{i-1}^1, u_i^j) + f_i^j & w_e E(u_{i-1}^1, u_i^j) & w_{mfcc} mfcc(u_{i-1}^1, u_i^j) \\ C_{i-1}^2 & w_f F(u_{i-1}^2, u_i^j) + f_i^j & w_e E(u_{i-1}^2, u_i^j) & w_{mfcc} mfcc(u_{i-1}^2, u_i^j) \\ \vdots & \vdots & \vdots & \vdots \\ C_{i-1}^m & w_f F(u_{i-1}^m, u_i^j) + f_i^j & w_e E(u_{i-1}^m, u_i^j) & w_{mfcc} mfcc(u_{i-1}^m, u_i^j) \end{Bmatrix} \\ \min\limits_k (C_n^k) \quad si \quad i = n+1 \end{cases} \tag{3.6}$$

$$p_i^j = arg(C_i^j) \tag{3.7}$$

tels que:m et k sont les nombres respectives des unités candidates d'ordres $i-1$ et n.

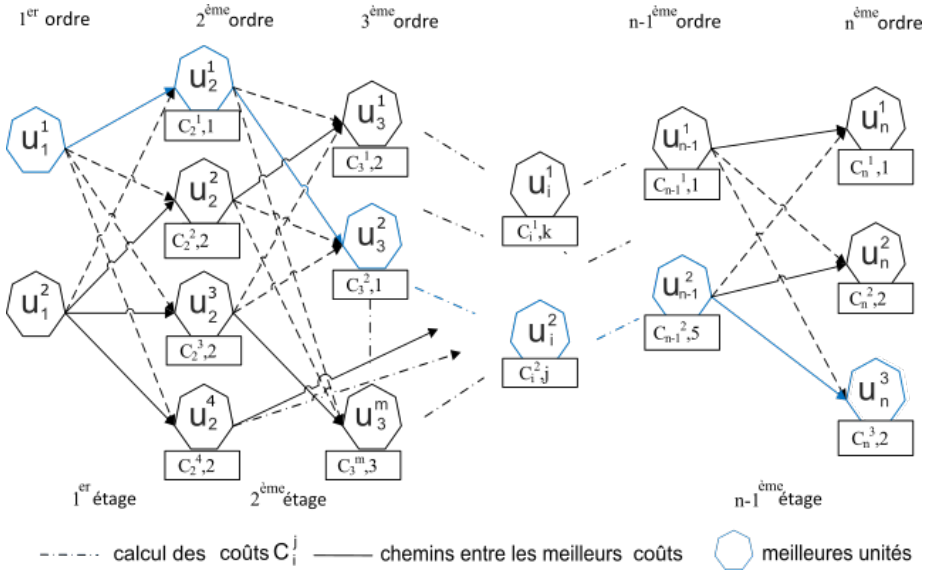

Figure 3.15: Recherche forward de la meilleure chaîne d'unités

3.4.2.2.2 Recherche Backward (RB)

Pour une meilleure sélection, une deuxième chaîne d'unités « Chaîne Backward (CB) » est générée avec une procédure similaire mais inverse par rapport à la première.

Dans cette seconde recherche, l'algorithme commence par les unités d'ordre n et se termine par celles d'ordre 1 (de droite à gauche) (figure 3.16). Contrairement à l'équation (3.6) le coût de concaténation est calculé avec les unités d'ordre supérieur, comme montre l'équation (3.8).

$$C_i^j = \begin{cases} 0 \quad si \quad i = n \\ min \begin{Bmatrix} C_{i+1}^1 & w_f F(u_{i+1}^1, u_i^j) + f_i^j & w_e E(u_{i+1}^1, u_i^j) & w_{mfcc} mfcc(u_{i+1}^1, u_i^j) \\ C_{i+1}^2 & w_f F(u_{i+1}^2, u_i^j) + f_i^j & w_e E(u_{i+1}^2, u_i^j) & w_{mfcc} mfcc(u_{i+1}^2, u_i^j) \\ : & : & : & : \\ C_{i+1}^m & w_f F(u_{i+1}^m, u_i^j) + f_i^j & w_e E(u_{i+1}^m, u_i^j) & w_{mfcc} mfcc(u_{i+1}^m, u_i^j) \end{Bmatrix} \\ \min_k(C_{n+1}^k) \quad si \quad i = 0 \end{cases} \tag{3.8}$$

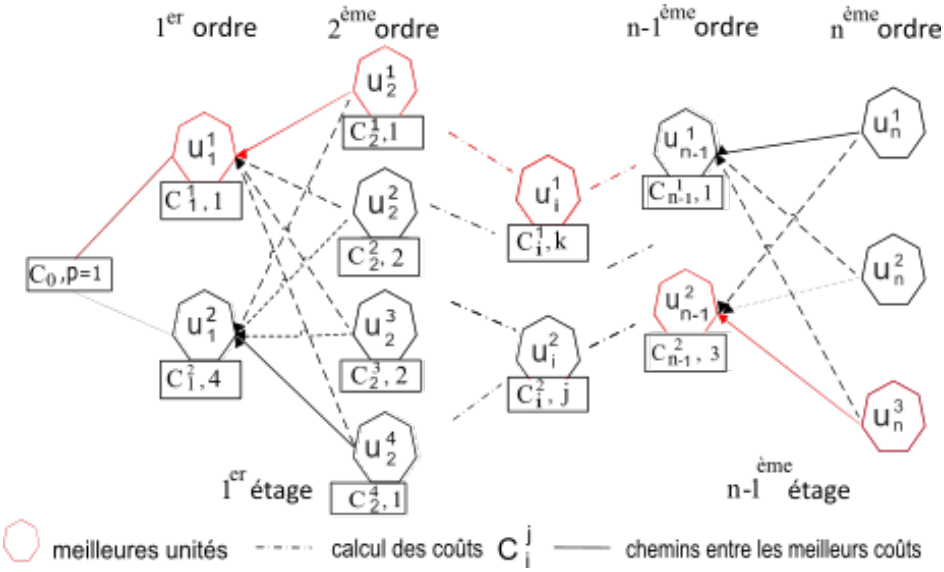

Figure 3.16: Recherche backward de la meilleure chaîne d'unités

3.4.2.2.3 Détermination de la chaîne finale

Après tous ces calculs et recherches, la chaîne finale d'unités se détermine en comparant les chaînes CF et CB. Dans chaque ordre, si les deux procédures ont sélectionné la même unité, cette dernière sera prise. Sinon l'algorithme rechoisit l'unité qui minimise la somme de son coût à gauche, trouvé par la RF, avec son coût à droite trouvé par la RB.

3.4.2.3 Sélection des unités pour les versets à lettres abrégées

Pour le cas des versets à lettres abrégées, nous voyons qu'une sélection contextuelle est suffisante. Car le nombre des unités dédiées pour ce type de versets est limité et toutes ces unités existent dans notre BD qui sont de 38 lettres (unités). Dans cette

phase, les meilleures unités à sélectionner $< L >$, sont celles qui ont les mêmes lettres (mots phonétiques) que le verset entré (cible). De plus ils maximisent le score de comparaison de trois caractéristiques contextuelles « position de la lettre (début, au milieu, ou à la fin), la lettre précédente et la lettre suivante » : S_p, S_d, S_g respectivement. Chaque terme de l'équation (3.9) reçoit le score « 1 » si les caractéristiques de l'unité cible et candidate s'accordent et « 0 » autrement.

$$< L >= \underset{j}{argmax} \left(S_d(t_j, l_i^j) + S_g(t_j, l_i^j) + S_p(t_j, l_i^j) \right) \tag{3.9}$$

tel que :

t_j : la j^{ème} unité cible ;

l_i^j : la i^{ème} lettre candidat pour la j^{ème} unité cible.

3.4.3 Concaténation des unités sélectionnées

Dans la phase de génération du son, les formes d'onde des unités sélectionnées se concatènent sans aucun traitement sur le signal résultant. Cette simple concaténation est adoptée pour éviter le problème de dégradation des sons au niveau de l'unité, dû au filtrage ou aux techniques de lissage d'un signal [71]. Notamment, la récitation du Coran nécessite la bonne prononciation de chaque phonème. À cause de la qualité d'enregistrement de certains sons dans la BD, une unification de la fréquence d'échantillonnage (22050 Hz ou 44100 Hz) et d'intensité de signal a été obligatoire.

3.5 Conclusion

Ce chapitre décrit l'élaboration d'un système de synthèse pour la récitation correcte du SC. Dans ce système, la SSU est adoptée comme méthode de synthèse, pour cela la construction d'une BD spécialisée a été nécessaire. Afin d'améliorer la qualité de la parole synthétique l'algorithme de sélection a été divisé en deux étapes successives. De plus, le principe des systèmes experts a été intégré par l'emploi des caractéristiques phonétiques et phonologiques de la langue arabe. L'évaluation de ce système (HQ_TTS), et la vérification de l'efficacité des approches proposées seront discutées dans le chapitre suivant.

CHAPITRE 4 : EVALUATION DU SYSTÈME HQ_TTS, RÉSULTATS OBTENUS ET DISCUSSIONS

4.1 Introduction

L'évaluation est une étape nécessaire dans l'élaboration d'un système de synthèse. Pour cela, se chapitre présente les tests effectués pour valoriser notre système HQ_TTS, les résultats obtenus ainsi que leurs discussions. Il s'agit des évaluations objectives et subjectives effectuées dans plusieurs points de vue. D'une part, nous allons vérifier la performance des principaux modules du système HQ_TTS. Et d'autre part nous allons tester la qualité de la parole synthétique, le temps nécessaire pour la synthèse et la récitation correcte du SC.

4.2 Performance des différents modules

Dans la première phase d'évaluation du système HQ_TTS, nous avons testé la performance de chaque module construit. Elle comprend des tests subjectifs et objectifs, effectués sous MATLAB à l'aide d'un ordinateur personnelle de processeur : Intel (R) Core (TM) i7-4600U CPU @ 2.10 GHz 2.70 GHz.

4.2.1 Transcription orthographique phonétique (TOP)

Le 1er chapitre a montré une diversité de méthodes d'évaluation des systèmes TTS de manière générale. Cependant, peu de travaux traitent l'évaluation de ces composants, notamment pour l'AS. La TOP est l'une des modules rarement évalués à cause du manque de standardisation des tests à effectuer. Il semble d'après [63, 72, 73], que la comparaison entre les résultats d'une transcription automatique et manuelle est la meilleure méthode d'évaluation.

Le test de la TOP du système HQ_TTS a été fait sur 45 phrases par le calcul des taux d'erreur de phonème Te_p et le taux d'erreur de mot Te_m. Le Te_p est calculé comme le pourcentage des phonèmes mal transcrits. Tandis que le Te_m correspond au pourcentage des mots ayant au moins un phonème erroné. Les versets de ce test ont été sélectionnés de manière à couvrir le maximum des cas d'exception et règles de transcription. Ils ont introduit au module de transcription comme un texte arabe entièrement diacritisé. Ensuite, le résultat pour chacun d'eux a été comparé avec une transcription manuelle déjà préparée.

Les résultats de cette comparaison donnent un Te_p = 100 % et Te_m = 100 %. D'une manière générale, les grandes erreurs de transcription se trouvent dans les abréviations, acronymes ou les mots étrangers à la langue. De plus, un texte en Arabe

souffre de la bonne transcription quand il n'est pas diacritisé. Ces problèmes sont limités dans notre système, ce qui prouve nos résultats.

Quelques exemples de transcription manuelle et automatique sont représentés respectivement dans le tableau 4.1 et la figure 4.1. De ces derniers, il est clair que tous les graphèmes et signes diacritiques ont été bien transcrits en leurs phonèmes correspondants. Ainsi que toutes les règles et exceptions ont été bien établies.

- les mots d'exception : « أُولَئِكَ » dans l'exemple 4 et « هَذَا » dans le 5ème exemple;
- la règle de « ال » de définition dans les exemples 1, 2, 3, 4 et 5 :« الْبَيْتِ » ;
- la règle de *hamzat-wasl* dans les exemples 6 « اسْتَغْنَى » et 8 « أْقرَأ » ;
- la règle de *alif maqsûra* dans les exemples 4 « هُدًى » et 6 « اسْتَغْنَى » ;
- l'élision de la voyelle et du *tanween* à la fin de la phrase comme dans l'exemple 3 « الْحُطَمَةِ » et 7 « هَاوِيَةٌ » respectivement ;
- les règles de *qalqalah* (exemple 1 « يَجْعَلُونَ »), *idgham* (exemple 4 « هُدًى مِنْ »), *ikhfaa* (exemple 2 « مِنْ شَرِّ ») et *iqlab* (exemple 3 « لَيُنبَذَنَّ ») ;
- les cas de *madd* : *lliin* (exemples 1 « الْمَوْتِ » et 5 « الْبَيْتِ »), *el-aarid li-ssukuun* (exemple 4 « الْمُفْلِحُونَ »).

Tableau 4.1: Transcription manuelle de quelques phrases de test

Le verset coranique	**La transcription manuelle**
يَجْعَلُونَ أَصَابِعَهُم فِي آذَانِهِمْ مِنَ الصَّوَاعِقِ حَذَرَ الْمَوْتِ	[#ja5_53aluuna!a$aabi3ahumfii+ !aa4aanihim-mina $$awaa3iqi 7a4ara lmaw+t#]
وَمِنْ شَرِّ النَّفَّاثَاتِ فِي العُقَدِ	[#wamin∼ carri nnaffaa8aati fil 3uqad_d#]
كَلَّا لَيُنبَذَنَّ فِي الْحُطَمَةِ	[#kallaa lajum'ba4anna fil 7u6amah#]
أُولَئِكَ عَلَى هُدًى مِنْ رَبِّهِمْ وَ أُولَئِكَ هُمُ الْمُفْلِحُونَ	[#!uulaa+!ika 3alaa hudan-min-rabbihim wa!uulaa+!ika humu lmufli7uu+n#]
فَلْيَعْبُدُوا رَبَّ هَذَا الْبَيْتِ	[#falja3buduu rabba haa4a lbaj+t#]
أَنْ رَآهُ اسْتَغْنَى	[#!an-ra!aahu stagnaa#]
فَأُمُّهُ هَاوِيَةٌ	[#fa!ummuhuu haawijah#]
أْقرَأ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ	[#!iq_qra! bismi rabbika lla4ii xalaq_q#]

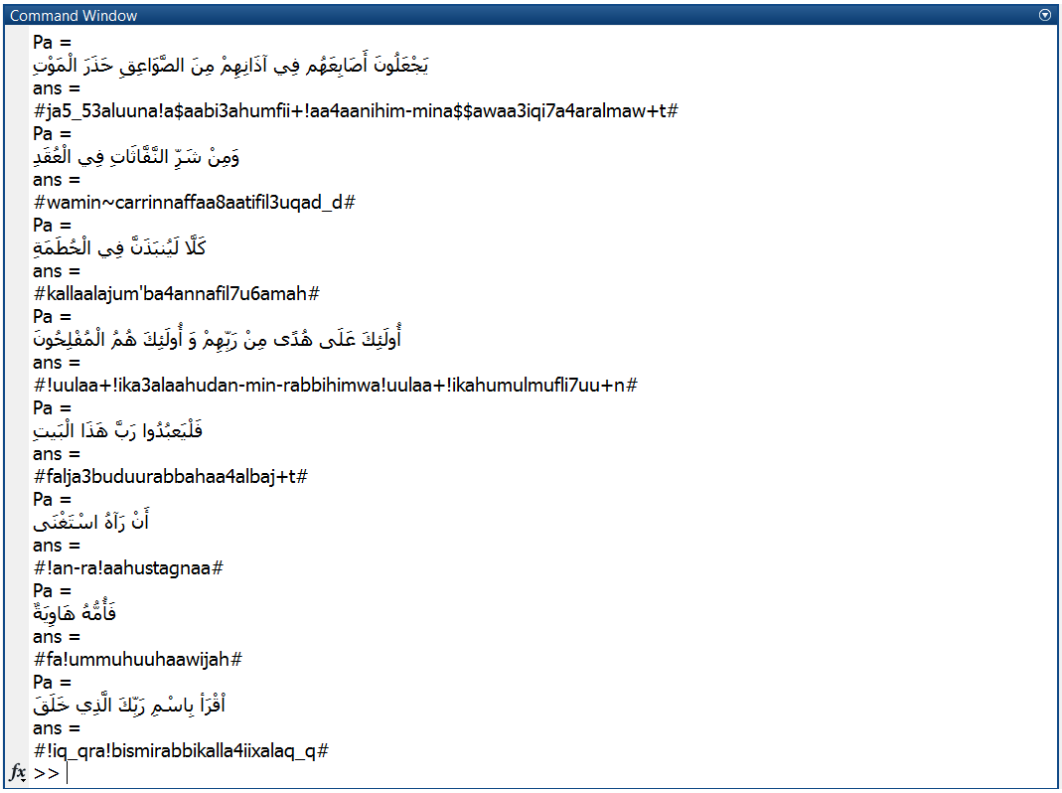

Figure 4.1: Résultat de la transcription automatique de quelques phrases de test

4.2.2 Procédure de sélection

L'évaluation de notre approche de sélection d'unités a été faite en deux étapes. Dans la première, nous avons évalué l'utilité de diviser le processus de sélection en deux algorithmes successifs. Dans ce contexte, nous avons pris 4 versets pour comparer le temps de synthèse de l'approche adopté avec celle usuelle comme expliqué dans le 2ème chapitre. Cette dernière fait la sélection des unités avec minimisation des coûts cible et de concaténation en même temps [28, 29, 74, 75]. Le résultat de cette comparaison est présenté dans la figure 4.2. De cette dernière, il est clair que notre approche est plus rapide, ce qui prouve la nécessité de cette subdivision d'algorithme de sélection.

Comme présente la figure 4.3, l'amélioration du temps de calcul ne dépend pas de la longueur de la phrase, mais du nombre des unités candidates dans le stade final de sélection. Dans cette dernière, le nombre des candidats est le même nombre initial issu de la BD au début de sélection par l'approche ordinaire. Par contre ce nombre diminue après la sélection contextuelle, si nous la séparons de l'autre. Cela veut dire que nous avons diminué le nombre d'opérations dans le calcul des coûts de concaténation. De ces résultats, nous conclurons que cette division de sélection est indispensable. Notamment quand nous allons appliquer une double recherche dans la phase finale de sélection, qui engendre un temps supplémentaire.

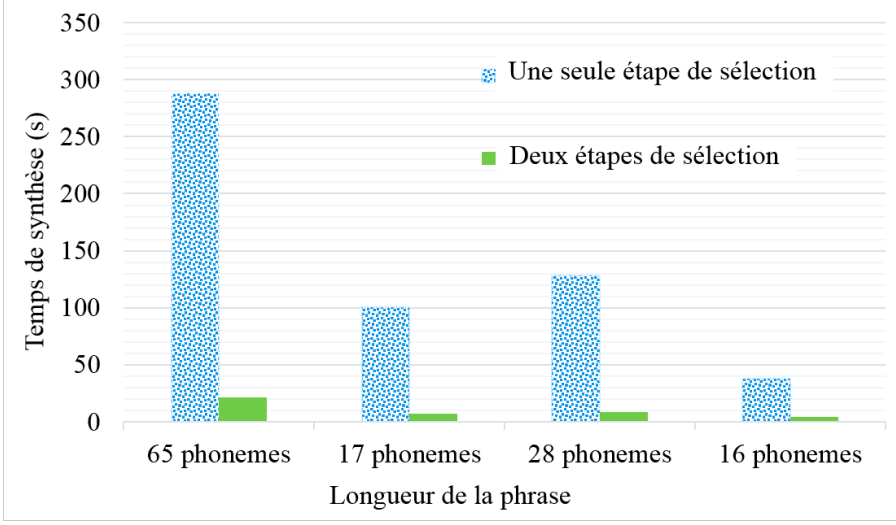

Figure 4.2: Temps de synthèse pour quelques phrases de test avec une et deux étapes de sélection

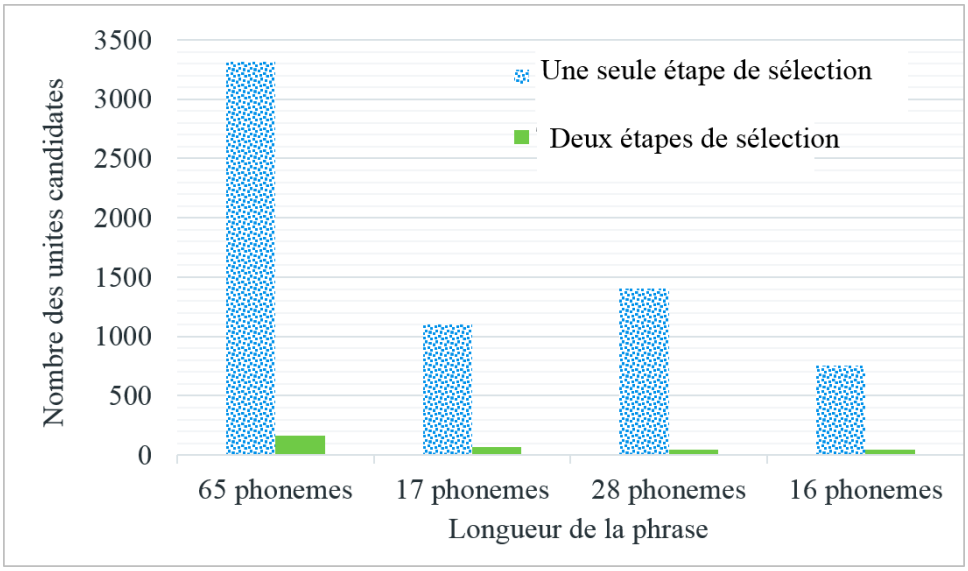

Figure 4.3: Nombre des unités candidates pour quelques phrases de test avec une et deux étapes de sélection

La deuxième évaluation a été faite pour étudier la double recherche forward-backward pour la sélection des unités. Pour cela, deux types de tests ont été réalisés. Le premier a été effectué pour étudier la différence entre les trois processus de recherche (Recherche Forward (RF), Recherche Backward (RB), et Recherche Combinée (RC)). Dans ce cas-là, nous avons comparé les trois chaînes d'unités résultantes de 26 différentes phrases. Comme présente le tableau 4.2, les chaînes issues du RF et RB diffèrent dans 30 % des cas synthétisés, et cette différence peut arriver jusqu'à cinq ou six unités. Nous trouvons aussi, que dans 38 % de ce résultat, la Chaîne Combinée (CC) ne corresponde à aucune des autres chaînes. Ce qui signifie que la RC peut sélectionner une unité qui n'a pas été choisie par les autres recherches (forward ou backward).

Tableau 4.2: Comparaison des chaînes obtenues par les trois processus de recherche forward, bakward et combinée

Cas de correspondance	**Nombre des phrases**	**Nombre des unités différentes dans les chaînes non correspondantes**
CF $\equiv$ CC	2	CB $\not\equiv$ dans [1 à 5] unités
CB $\equiv$ CC	3	CF $\not\equiv$ dans [1 à 2] unités
CF $\equiv$ CC $\equiv$CB	18	/
CF $\not\equiv$ CC $\not\equiv$ CB	3	CF $\not\equiv$ CC dans [1 à 4] unités

Dans le deuxième test, nous avons synthétisé 10 différentes phrases en utilisant les trois processus de recherche. Ces phrases ont été choisies de telle sorte que leurs chaînes forward et backward diffèrent en trois unités au moins. Sept personnes ont effectué cette évaluation en écoutant les trois versions de chaque phrase, après ils choisissent la version qu'ils préfèrent en termes de qualité de la parole. Le résultat de ce test donne une préférence de 58 % des phrases synthétisées par la recherche combinée, malgré les difficultés qu'ils avaient les évaluateurs pour différencier entre les trois versions (15 % des cas, les phrases ont été marquées par les auditeurs comme étant les mêmes). Cette difficulté de distinction est compréhensible, puisque la différence entre les trois chaînes n'était pas assez grande dans certains cas, et généralement, les unités différentes sont un peu espacées entre eux (pas successives). Pour cette raison, il n'est pas facile pour une oreille humaine de faire la difference.

4.2.3 Ajustement des scores

La dernière évaluation des modules de HQ_TTS a été faite pour tester notre approche d'introduire les SE dans la sélection des unités, plus exactement pour la pondération des caractéristiques. D'abord, nous avons vérifié l'utilité d'ajustement des poids (ou score) pour la bonne sélection des unités. Dans ce cas-là, par un test subjectif nous avons comparé la qualité de la parole synthétique d'une sélection basée sur des techniques existantes d'ajustement des poids : l'AG, l'AGai et la technique basique (ou standard) qui est sans pondération). Dans cette dernière, chaque caractéristique d'unité reçoit un score binaire : «1 » s'il y a une correspondance entre l'unité cible et candidat et « 0 » dans le cas contraire). 15 auditeurs ont participé à cette évaluation, où ils sont invités d'écouter 10 versets synthétisés en utilisant les trois techniques précédemment citées. Ensuite, ils ordonnent les trois versions de chaque exemple en termes de qualité de la parole [76].

La figure 4.4 présente le résultat de cette comparaison. Elle montre la préférence de l'AGai pour la meilleure qualité de la parole, où elle été classée comme un premier

choix dans 41.17 % des cas. Son résultat a été mieux que l'AG ce qui prouve l'utilité de l'interaction humaine dans le processus d'ajustement des poids. La version standard est la moins préférée ce qui confirme la nécessité d'une pondération entre les caractéristiques d'unité.

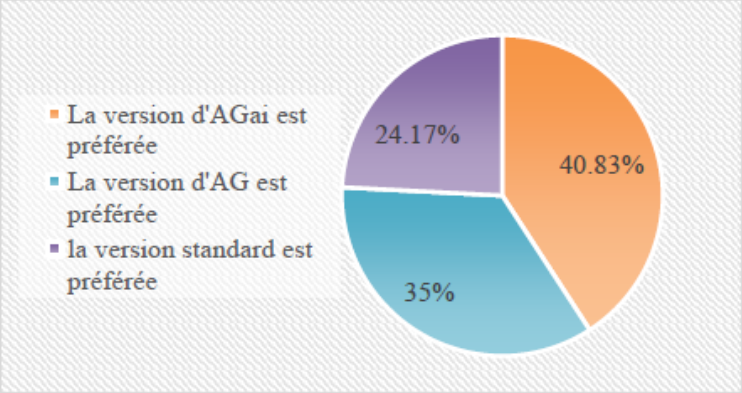

Figure 4.4: Résultat de comparaison de qualité de la parole de trois techniques de pondération des caractéristiques

Après ce test vérificatif nous avons comparé notre approche des SE par rapport à la technique basée sur l'AGai. En plus des versets précédents, cinq autres sont aussi utilisés. Dans ce test, nous avons demandé aux auditeurs d'évaluer la qualité de parole pour chacune des versions dans une échelle de 1 (très mauvaise) à 5 (très bonne) [77, 78]. À la fin la version ayant le meilleur score moyen est la plus préférée. Le résulté de ce test est présenté dans la figure 4.5. Elle montre une claire préférence de l'approche de SE par 54 %. Ce pourcentage est nettement mieux que 36.2 % pour aucune préférence ou 9.8 % pour l'AGai, qui est un résultat encourageant.

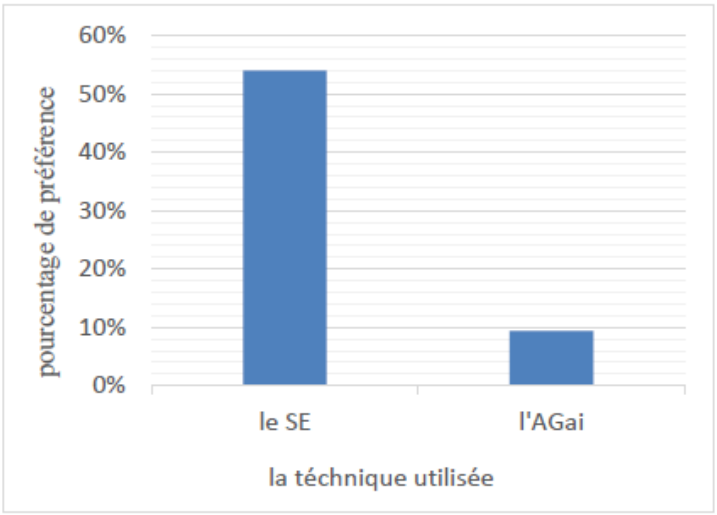

Figure 4.5: Résultat de préférence de la qualité de la parole entre la synthèse avec : le SE et l'AGai

L'équation (4.1) présente la complexité de notre algorithme de recherche forward-backward. Nous constatons à partir de cette équation qu'un petit nombre d'unités

candidates (issues de la sélection contextuelle) réduit le temps de synthèse. Pour cela, nous avons aussi comparé le nombre des unités sélectionnées (après le calcul du score cible) par le SE, l'AGai et la Technique Standard (TS). La figure 4.6 présente le résultat de quelques phrases de test. Elle montre que le nombre d'unités sélectionnées par l'ES_US est significativement inférieur par rapport aux autres méthodes, quelle que soit la longueur de la phrase, ce qui prouve son efficacité de sélection.

$$CO = 2 \times \sum_{i=1}^{n} k_i \times k_{i+1} \tag{4.1}$$

tel que :

n : est le nombre des unités (diphones) constituants la phrase à synthétiser ;

k_i : est le nombre des candidats pour la i ème unité cible.

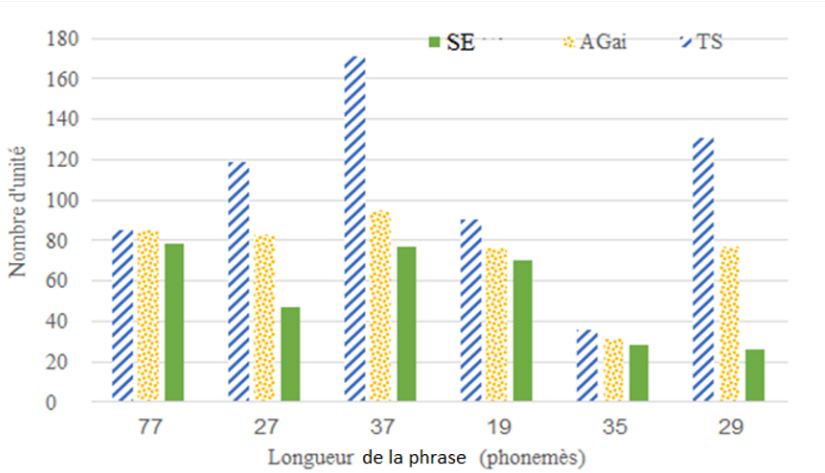

Figure 4.6: Nombre d'unités sélectionnées par la fonction cible en utilisant trois approches de pondération des scores

4.3 Qualité de la parole

La qualité de la parole synthétisée a été évaluée par l'application de deux tests subjectifs. L'un pour l'intelligibilité et l'autre pour le naturel de la parole. Cette évaluation a été effectuée on-line [79, 80] pour avoir un maximum nombre de participants. À la fin et après filtration nous avons pris les résultats de 90 auditeurs musulmans ; âgés entre [15 et 50] de différentes nationalités et origines linguistiques.

4.3.1 Intelligibilité

Dans le premier test d'intelligibilité de la parole, 10 phrases (versets) et 20 mots ont été utilisés. Les auditeurs devaient écouter à chaque Phrase ou Mot (P / M) et récrire ce qu'ils avaient entendu. Entretemps, nous notons intelligible s'ils reconnaissent très

bien les principaux phonèmes de la P / M cible et non intelligible dans le cas contraire. Les réponses ont été bien traitées pour enlever l'ambigüité due aux erreurs de frappe ou d'orthographe, qui ne nous s'intéressent pas. Les versets choisis comprennent : les deux phrases principales dans la récitation du SC (*basmalah* et *istiadhah*), une paire de versets courts et similaires avec deux phonèmes différents (l'exemple 3 et 4 : العَزِيزُ الحَكِيمْ et العَزِيزُ الرَحِيمْ) et d'autres versets de différentes longueurs et variétés phonétique et phonologique (tableau 4.3). Les mots utilisés dans ce test ont été aussi choisis avec différentes longueurs et variétés phonèmes (tableau 4.4). À la fin, un pourcentage d'intelligibilité I (équation 4.2) a été calculé comme le nombre de P / M intelligibles devisé par le nombre total. L'équation (4.3) présente le calcul de l'intervalle d'erreur E, qui est mis pour les cas où nous ne sommes pas sûrs si l'auditeur a fait une erreur de frappe ou il n'a pas bien reconnu la phrase / mot.

$$I = \frac{nombre\ des\ P/M\ nots\ inteligible + \frac{1}{2}\ du\ nombre\ des\ P/M\ suspectibles}{Nombre\ totale\ des\ P/M} \tag{4.2}$$

$$E = \frac{\frac{1}{2}\ du\ nombre\ des\ P/M\ suspectibles}{Nombre\ totale\ desP/M} \tag{4.3}$$

Le résultat de ce test indique que les phrases synthétisées donnent un pourcentage d'intelligibilité globale de phrase de 96.21 ± 2.57 % et 86.56 ± 6.44 % pour les mots. Les tableaux 4.3 et 4.4 présentent les résultats détaillés de chaque mot et phrase. Les auditeurs commentent, en général, qu'ils n'ont pas eu de difficultés pour reconnaître les phrases synthétisées. Les résultats du test ne sont pas parfaits à cause de la non standardisation d'écriture des auditeurs qui nous mise en ambiguïté pour bien classer les phrases. Par exemple, dans la 2 ème phrase y a des gens qui l'écrivent comme ils entendent « بِسْمِ الله الرَّحمَانِ الرَّحِيمْ » sans hamzat wasl, qu'elle ne prononce pas, et d'autres comme « بِاسمِ اللهِ الرَّحمَانِ الرَّحِيمْ » , sans parler du manque des signes diacritiques. Un autre défi, a été comment interpréter et écrire les changements dus aux règles de *tajweed*. Le plus grand problème d'intelligibilité de phrase a été dans l'exemple 7, où certains auditeurs ont confus la prononciation du ذ [4] par ث [8]. Ces deux phonèmes se ressemblent dans leur friction et point d'articulation, et peuvent se confondent à cause de l'effet de l'outil d'évaluation (casque, haut de parleur, etc.) ou bien de la non exactitude du contexte de l'unité sélectionnée par rapport au contexte cible.

Pour les mots synthétisés le tableau 4.4 indique que la plupart des mots ont été bien reconnu sauf :

Tableau 4.3: Pourcentage d'intelligibilité des phrases synthétisées

Verset	**Transcription en code personnel**	**Pourcentage d'intelligibilité**
أَعُوذُ بِاللَّهِ مِنَ الشَّيطَانِ الرَجيمِ	[#!a3uu4u billahi mina ccaj6aani rra5ii+m#]	99.42±0.57%
بِسمِ اللهِ الرَّحمَانِ الرَّحِيمِ	[#bismi llahi rrqhmaani rra7ii+m#]	95.4±4.59%
العَزِيزُ الحَكِيم	[#!al3aziizu l7akii+m#]	100%
العَزِيزُ الرَّحِيم	[#!al3aziizu rra7ii+m#]	100%
إِنَّا أَنزَلنَاهُ فِي لَيلَةِ القَدرِ	[#!innaa+ !anzalnaahu fii lajlati lqad_dr#]	97.02±2.97%
فَالمُورِيَاتِ قَدحَا	[#falmuurjaati qad_d7aa#]	96.99±1.81%
كَلَّا لَيُنبَذَنَّ فِي الْحُطَمَةِ	[#kallaa lajum'ba4anna fil 7u6amah#]	89.76±1.81%
آمَنَ الرَّسُولُ بِمَا أُنزِلَ إِلَيهِ مِن رَبِّهِ وَ المُؤمِنُون	[#!aamana rrasuulu bimaa+ !unzila !ilajhi min-rrabbihi walmu!minuu+n#]	92.94±3.53 %
اللَّهُ الَّذِي رَفَعَ السَّمَوَاتِ بِغَيرِ عَمَدٍ تَرَونَهَا	[#!allahu lla4ii rafa3a sab3a samaawaatin bigajri 3amadin∼ tarawnahaa#]	91.76±2.35%
وَأَخَذْنَا مِنْهُم مِيثَاقًا غَلِيظَا	[#wa!axa4naa minhum mii8aaqan galii%aa#]	98.84±1.16%

- إِيتَاءْ [!iitaa!] où la reconnaissance du phonème [!] qui pose du problème. Ce dernier est dû à la petite durée et le caractère plosif de ce phonème qui empêche son identification au début de la phrase dans certains contextes. Nous constatons aussi que ce même phonème est mieux reconnu s'il est suivi par une consonne silencieuse (le cas des mots إِنَّ [!inna] et إِبْتِغَاء [!ib_btigaa!] ;

- يَبْخَسْ [jabxas] et قِرْطَاس [qir6aas], l'analyse de transcription de ces mots résulte que le son du س [s] peut-être confus avec le ص [$] dans certains cas. Ces deux phonèmes partagent le même point d'articulation et les caractéristiques de friction et de voisement et autres. La seule différence est l'effet emphatique engendré avec le [$]. Pour cela, nous concluons que dans ces exemples l'emphase des phonèmes voisins du [s] a influencé sur ses caractéristiques, où certains auditeurs la perçoivent comme [$]. Cette conclusion peut être confirmée par le taux d'intelligibilité du mot المُسْتَغفِرِينْ [!almustagfirii+n], ou le phonème [s] a été bien identifié ;

- قِسْط [qis6_6] dans certains cas cet exemple a été identifié comme [qis6a]. Ceci par ce que les auditeurs ne savent pas comment représenter ou interpréter la *qalqalah* et ils la perçoivent comme [a]. Il été indiqué aussi dans la thèse

de Ahmed Raghib [10] que les caractéristiques acoustiques de qalqalah et le phonème [a] se ressemble un peu, ce qui explique cette perception ;

- ثَجَّاجَا [8a55aa5aa] et لُجِّي [lu5ijjin]: ici, l'écoute des mots hors contexte grammatical, surtout s'ils sont rarement utilisés, complique leur identification. Ce qui mène les évaluateurs de transcrire le plus proche son ou mot qu'ils reconnait dans certains cas. Aussi la différence de prononciation du ج [5] dans les différents dialectes arabes peut influencer la bonne perception de celle-ci surtout quand elle est géminée ou en mode de qalqalah. Cela est peut être dû à la mal connaissance des règles de *tajweed* par les auditeurs ce qui empêche la bonne indentification de ce phonème.

Tableau 4.4: Quelques mots synthétisés et leurs pourcentages d'intelligibilité

Mot en Arabe	**Transcription**	**Pourcentage d'intelligibilité**
مُدْهَامَّتَانْ	[#mudhaaammataa+n#]	84.62 ± 2.56%
ص	[#$aaad#]	100%
يَبْخَسْ	[#jab_bxas#]	85.5%
إِنَّ	[#!innah#]	95.45%
أَشِحَّةً	[#!achi77atan#]	83.55 ± 0.66%
يَتَوَكَّلُونْ	[#jatawakkaluu+n#]	100%
إِيتَاءْ	[#!iitaa !#]	66.67%
يُوحَى	[#juu7aa#]	100%
نَظِرَه	[#na%irah#]	75.64 ± 9%
فَضْلِهْ	[#fa£lih#]	95.12 ± 2.3%
بَعْضَهُمْ	[#ba3£ahum#]	95.12 ± 2.3%
الشُّهَدَاء	[#!accuhadaa!#]	96.43 ± 1.2%
صَرَّفنَا	[#$arrafnaa#]	100%
لُجِّي	[#lu5ijjin#]	31.25%
مَلَكَتْ	[#malakat#]	100%
المُسْتَغفِرِينْ	[#!almustagfiriin#]	100%
إِبْتِغَاء	[#!ib_btigaa! #]	100%
طَائِرًا	[#6aa !iraa#]	52.63 ± 23.68%
فَتُذَكِّرْ	[#fatu4akkir#]	97.62%
مَوعِظَه	[#maw3i%ah#]	73.8 ± 0.21%
قِرطَاس	[#qir6aas#]	50 %
قِسْط	[#qis6_6#]	68.18 ± 4.5 %
رِزْقِه	[#rizqih#]	100%
ثَجَّاجَا	[#8a55aa5aa#]	30 ± 3.33 %

Nous remarquons aussi que la non existence de l'exact contexte cherché dans la BD

fausse la bonne identification de certains phonèmes, où ils se perçoivent comme d'autres proches de caractéristiques : le ط [6] comme ت [t] ; le ث [8] comme ف [f], etc.

Ces petits problèmes d'intelligibilité des mots non pas forcément liés à l'efficacité du système de synthèse, comme les systèmes SSU non pas de problème d'intelligibilité en général. La qualité d'enregistrement des sons utilisés et l'outil utilisé pour l'évaluation jouent aussi un rôle très important pour la reconnaissance des sons. Heureusement, ces problèmes n'affectent pas beaucoup l'intelligibilité de la phrase car ils sont des cas rares et un verset coranique peut être prédit même avec un phonème non identifié.

4.3.2 Naturel

Dans le deuxième test du naturel, les auditeurs ont écouté 20 phrases (versets) synthétisées. Puis ils ont évalué leur satisfaction sur le naturel de la parole dans une échelle de [1 á 5] (Très mauvais, Mauvais, Moyen, Bien, Très bien). Ensuite, le MOS a été calculé pour chaque verset. Ces versets ont été choisis parmi les deux types mentionnés dans le chapitre précédent avec des longueurs différentes, dans des contextes distincts et avec différents pourcentages d'utilisation dans l'élaboration de la BD (certains versets ont été utilisés à 100 % d'autres à 50 %, 30 % ou ils ne sont pas utilisés de tout) (tableau 4.5).

Ce test a donné un résultat global de 73.2 % de naturel (un MOS de 3.66). Comme présente les figures 4.7 (a), (b) et (c), le résultat détaillé de ces versets donne de bons et moyens scores de parole naturelle. De ces valeurs, nous constatons que le système HQ_TTS synthétise très bien les versets utilisés pour la construction de la BD (comme les versets 2, 7 et 9), cela veut dire qu'il a trouvé les exactes unités cherchées. De plus, il synthétise mieux les versets courts (les versets 4 et 5 par rapport à la 1), car plus la phrase est longue, plus l'effet de transition entre les unités est perçu et devrait ennuyant. Les résultats où le naturel du verset a un score inférieur à «3», sont dus au petit nombre de variantes de quelques unités rares dans la BD , comme : [xxaa] , [iiz] , etc.). La qualité d'enregistrement des sons utilisés a également un effet important. Certains des sons essentiels étaient mauvais et un peu bruités à l'origine. À ce jour, ils représentent la seule source d'un enregistrement monotone du SC. Comme l'exemple14, ce problème affect beaucoup les courts versets où une seule mauvaise unité peut dégrader la qualité de toute la phrase.

Tableau 4.5: Phrases utilisées dans le test de naturel de la parole et leurs pourcentages d'utilisation dans la BD

N° phrase	Verset	Pourcentage d'utilisation dans la BD
1	الْحَمْدُ لِلَّهِ الَّذِي أَنزَلَ عَلَى عَبدِهِ الْكِتَابَ وَلَمْ يَجعَل لَهُ عِوَجَا	0%
2	وَلَهُم عَذَابٌ أَلِيمٌ بِمَا كَانُوا يَكذِبُون	100%
3	يَحسَبُ أَنَّ مَالَهُ أَخلَدَهُ	24%
4	يَا أَيُّهَا الَّذِينَ آمَنُو	0%
5	هَل أَتَاكَ حَدِيثُ الْغَاشِيَةِ	0%
6	فَلْيَعْبُدُوا رَبَّ هَذَا الْبَيْتِ	21%
7	أُولَئِكَ عَلَى هُدًى مِنْ رَبِّهِمْ وَأُولَئِكَ هُمُ الْمُفْلِحُونَ	100%
8	قُل يَا أَيُّهَا الْكَافِرُون	30%
9	ذَلِكَ الْكِتَابُ لَا رَيبَ فِيه	100%
10	قُل أَعُوذُ بِرَبِّ الْفَلَق	95%
11	عَمَّ يَتَسَاءَلُون	0%
12	وَالضُّحَى وَاللَّيلِ إِذَا سَجَى	0%
13	أَرَأَيتَ الَّذِي يُكَذِّبُ بِالدِّين	44%
14	أَنْ رَآهُ اسْتَغْنَى	0%
15	كهعص ذِكرُ رَحمَةِ رَبِّكَ عَبدَهُ زَكَرِيَّا	30%
16	خَلَقَ الإِنسَانَ مِنْ صَلْصَالٍ كَالفَخَّار	50%
17	إِقرَأ بِسمِ رَبِّكَ الَّذِي خَلَق	80%
18	والسَّمَاءِ وَالطَّارِق	0%
19	سَبِّحِ اِسْمَ رَبِّكَ الْأَعلَى	0%
20	فَأُمُّهُ هَاوِيَةٌ	90%

Donc par l'utilisation des sons bien enregistrés et l'enrichissement de la BD, la qualité du système sera améliorée. La variance des résultats pour chaque phrase peut s'expliquer par les attentes élevées de certains évaluateurs. Tout d'abord, ils ne sont pas familiarisés avec une parole synthétique. Deuxièmement, certains d'entre eux comparaient la qualité des versets récités par le système HQ_TTS à ceux qu'ils avaient l'habitude d'entendre. Ces derniers sont non seulement bien enregistrés mais aussi prononcés avec un style de récitation artistique et une mélodie agréable par rapport au style monotone de HQ_TTS. Heureusement ce mauvais résultat n'est pas général et se pose dans certaines minorités de phrases, que nous avons testé certains parmis eux pour voir le plus mauvais score que le système peut obtenir. Pour valider cette dernière hypothèse, un test de naturel à choix libre des phrases a été aussi effectué.

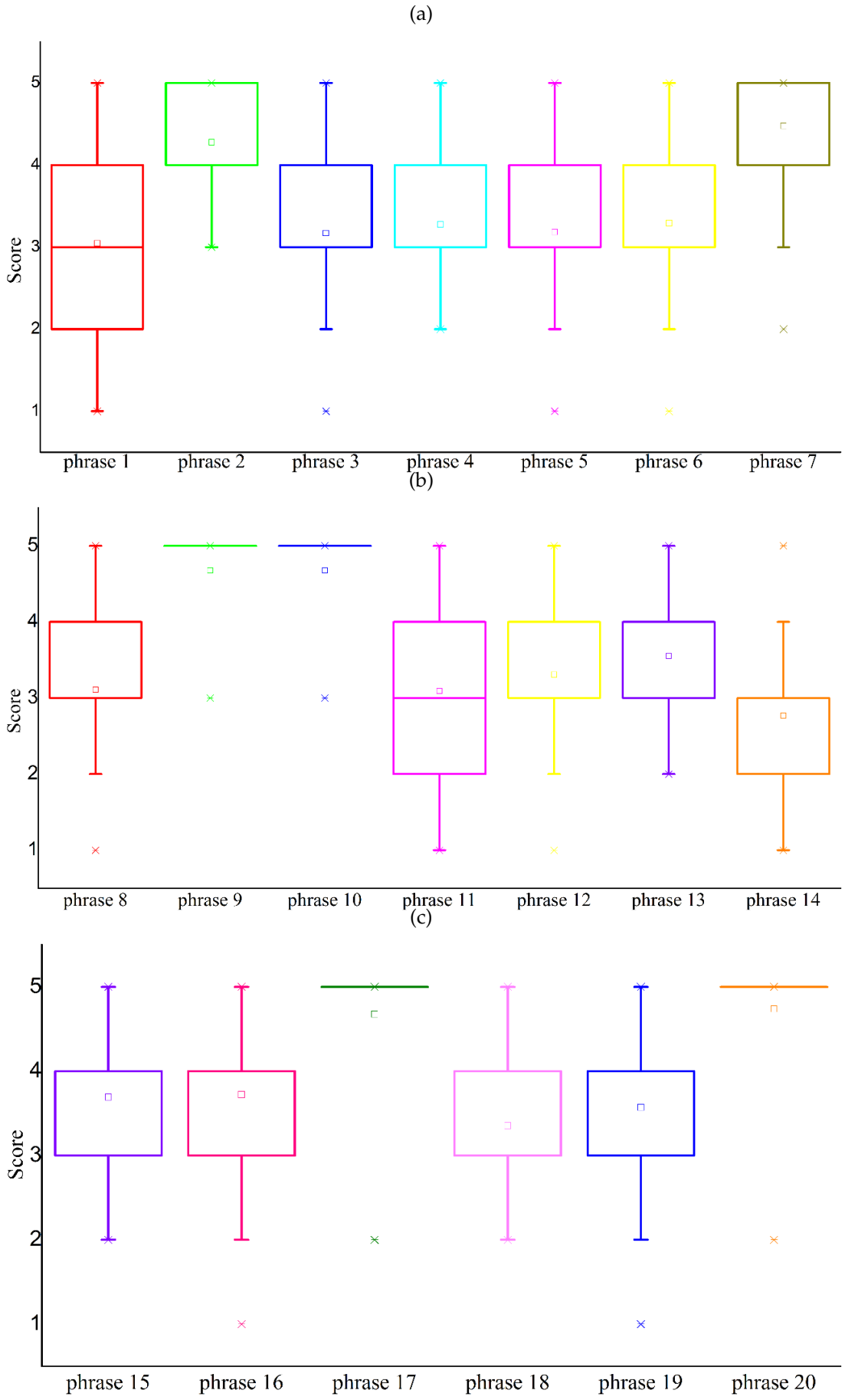

Figure 4.7: Boîte à moustaches des scores de test du naturel de la parole synthétique

À l'aide d'une interface graphique développée sous MATLAB (figure 4.8). 20 personnes ont été évalués le naturel de quatre versets à leur choix.

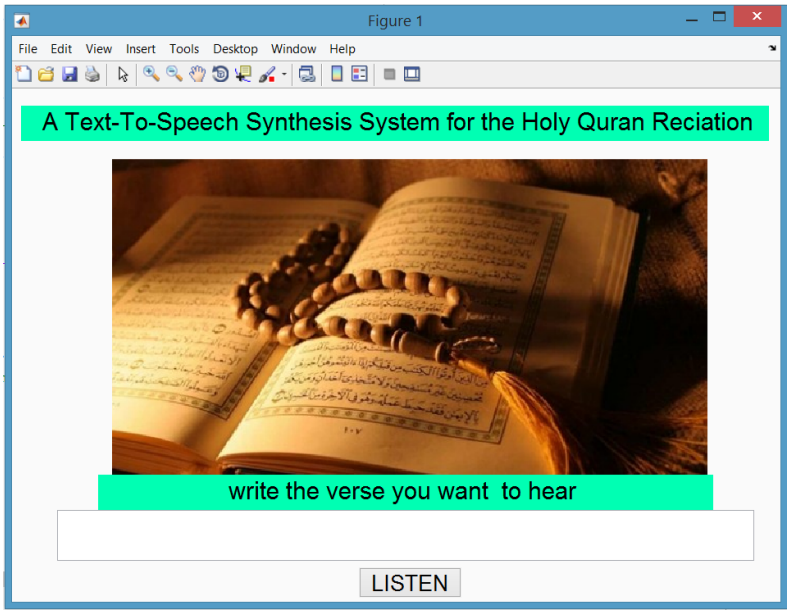

Figure 4.8: Interface d'utilisation du système HQ_TTS

Avec ce test, 70 versets et parties de versets ont été synthétisés. Comme présente le tableau 4.6, le résultat est très encourageant où plus de 63 % des phrases ont un score supérieur à 4).

Tableau 4.6: Résultat du naturel de la parole synthétisée pour les phrases à choix libre

Score	**1**	**2**	**3**	**4**	**5**
Pourcentage des phrases (%)	1.93	7.67	26.7	33.5	30.2

4.4 Evaluations générales

En plus de l'évaluation de qualité de la parole synthétisée et la performance des modules élémentaires du HQ_TTS, d'autres tests ont été nécessaires pour valider le système conçu.

Parmi ces tests, nous avons évalué la récitation correcte du système HQ_TTS en utilisant la même plateforme du test de qualité vocale précédent. Pour cela, nous avons Interrogé que les auditeurs à bonne ou moyenne connaissance des règles de *tajweed*. Après qu'ils écoutent et analysent les versets synthétisés, ils choisissent entre « une très bonne récitation (pas d'erreur)», « existence de quelques erreurs » ou bien « beaucoup d'erreurs (les règles n'ont pas bien respecté) ».

Le résultat de ce test confirme la bonne récitation de notre système où 85 % des choix ont été pour « une très bonne récitation (pas d'erreur) ». Le reste du pourcentage (15 %) était pour « existence de quelques erreurs », ou quelques participants étaient confus avec certaines règles de *tajweed* dans certains cas. D'après l'analyse de ces résultats et les commentaires des auditeurs, nous constatons que, parfois, la qualité

de la parole affecte la bonne réalisation de ces règles, plus exactement, dans le cas du *madd* (sa longueur exacte) et l'assimilation entre le [r] et le [n].

De point de vue subjectif, durant les tests réalisés, nous avons demandé aux auditeurs de commenter et donner leurs avis sur le système HQ_TTS et son temps de réponse, sa vitesse et compréhensibilité, ainsi que leur acceptation générale de la qualité de la parole synthétisée. Les réponses pour le système ont été encourageantes pour plus de développement et amélioration. La parole synthétisée a été compréhensible et de vitesse très acceptable. La qualité globale de la synthèse a donné un score de 3.7 sur 5. Les évaluateurs dans ce cas ont été ennuyés par la discontinuité des sons de certains versets de test et ils demandent l'amélioration de ça.

Pour compléter nos évaluations du système HQ_TTS une analyse du temps de synthèse a été effectuée. Cette analyse est faite à l'aide de la fonction « profile » de MATLAB. La figure 4.9 montre le profil d'exécution de ce système, en synthétisant le verset « قُل أَعُوذُ بِرَبِّ الْفَلَقْ ».

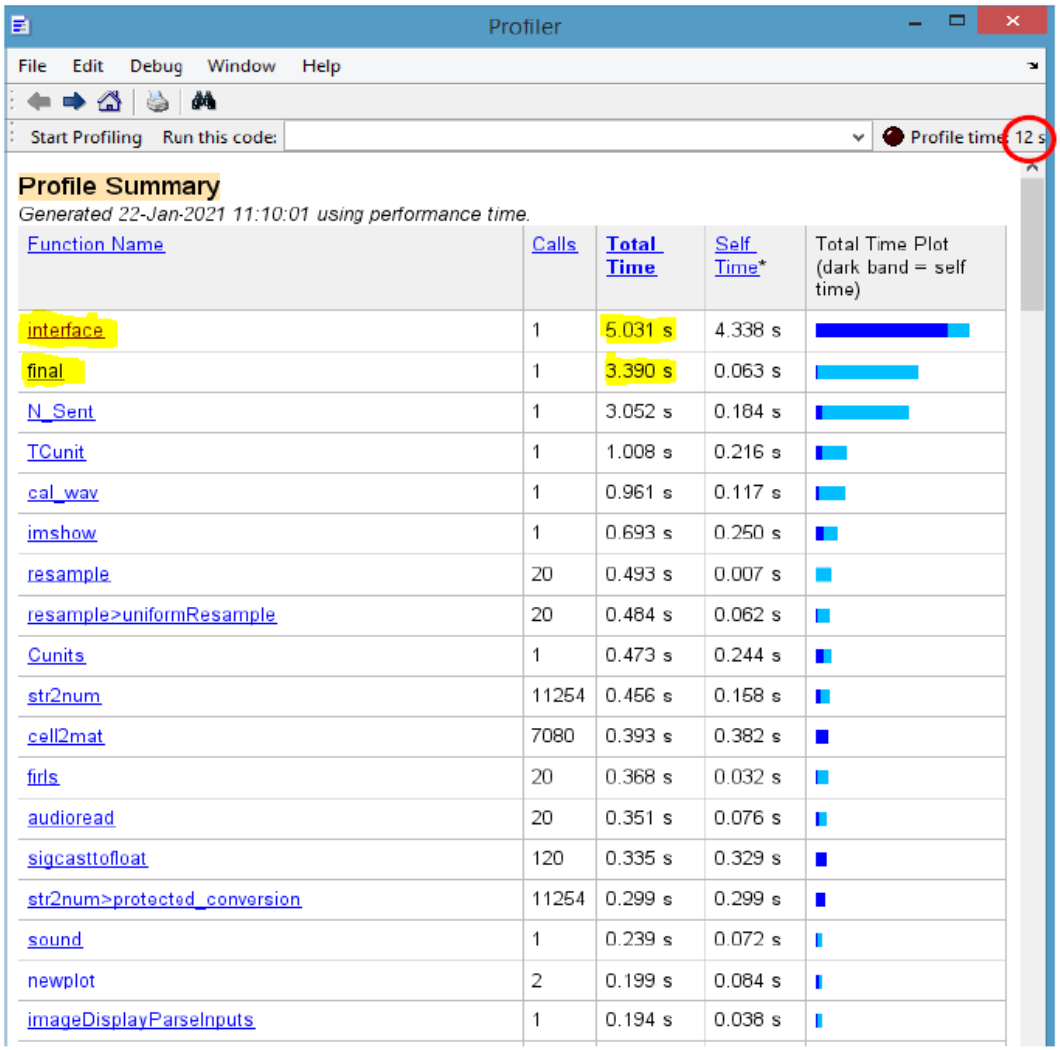

Figure 4.9: Profil d'exécution du système HQ_TTS

La figure 4.9 indique que le temps d'exécution de HQ_TTS pour ce verset est de 12 s. Cependant, notre programme fait l'appel à deux fonctions principales « *interface* » et « *final* ». Ces fonctions consomment respectivement 5.031s et 3.390 s. Par conséquent, le temps réel de la synthèse est d'environ 8.421 s. Le reste du temps, 3.579 s, a été consommé par l'utilisateur durant la saisie du verset dans l'interface présentée dans la figure 4.8. L'analyse détaillée de ce profil montre que la moitié de la durée d'exécution est consomée par le chargement de la BD. Ce dernier est l'une des fonctionnalités optionnelles de la fonction « *interface* » qui déroule pendant 4.3 s (figure 4.10). Nous avons Noté que ce temps est presque invariable pour déférents exemples. Il dépend seulement de de la rapidité et la capacité de la machine qui effectue le calcul. D'une manière générale, ce temps est acceptable, car le chargement de BD se fait qu'une seule fois au premier lancement du programme.

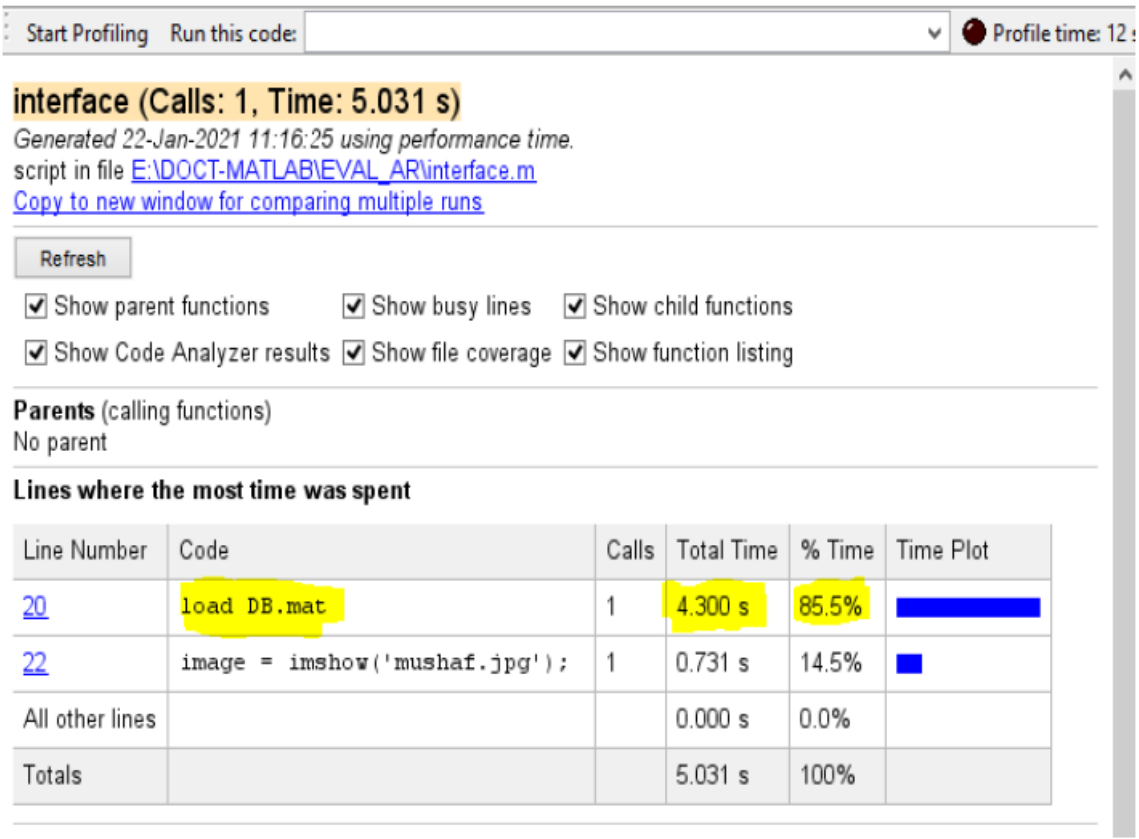

Figure 4.10: Profil d'exécution de la fonction « *interface* » du système HQ_TTS

Le tableau 4.8 montre la consommation temporelle des différents modules pour les phrases suivantes (tableau 4.7).

D'après ce tableau de consommation, nous constatons que :

- le temps de transcription et d'annotation est presque le même pour tous les exemples. Il dépend un peu de la longueur de phrase (comme l'exemple 6), mais principalement de sa complexité linguistique, où plusieurs règles de

Tableau 4.7: Phrases de test de la consommation temporelle du système HQ_TTS

N°	Verset	Transcription	Longueur (phonèmes)
1	قُل يَا أَيُّهَا الْكَافِرُون	[#quljaa!ajjuhalkaafiruu+n#]	21
2	مُدهَامَّتَان	[#mudhaaammataa+n#]	12
3	قُل أَعُوذُ بِرَبِّ الْفَلَق	[#qul!a3uu4ubirabbilfalq_q#]	23
4	قُل هُوَ اللَّهُ أَحَد	[#qulhuwallahu!a7ad_d#]	18
5	عَمَّ يَتَسَاءَلُونَ	[#3amma jatasaa!aluu+n#]	17
6	قَد سَمِعَ اللَّهُ قَولَ اللَّتِي تُجَادِلُكَ فِي زَوجِهَا	[#qad_dsqmi3allahuqawlallatii tu5aadilukafiizaw5ihaa#]	43
7	إِنَّ اللَّهَ بِالنَّاسِ لَرَؤُوفٌ رَحِيم	[#!innallahabinnaasilara!uufun-ra7ii+m#]	29
8	إِذَا زُلزِلَتِ الْأَرضُ زِلزَالَهَا	[#!i4aazulzilatil!ar£uzilzaalahaa#]	30

transcription ont mis en jeu. Par exemple les versets 3 et 7 ont un nombre de phonèmes inférieure ou égale au verset 8, mais leurs temps de transcription est relativement supérieur. L'analyse de ce verset montre que sa transcription est une conversion directe des caractères sans règles spéciales, ce qui confirme son temps de transcription réduit ;

- la première sélection prend le plus grand temps de calcul. Ceci est logique, car la partie consommatrice du temps est la recherche des unités dans la BD et le calcul de fonction cible. Ce temps dépend de la longueur de la phrase et surtout le nombre d'unités candidats (nombre des variétés pour chaque unité cible). Ce point nous renvoie à l'étape de construction de la BD où nous devons bien étudier et choisir le nombre et le type des variétés qui doivent s'ajouter pour chaque diphone ou polyphone (un compromis entre une variété contextuelle et limitation de la taille de la BD doit se fait). L'utilisation des filtres de présélection, comme présenté dans le travail de Gunnec [25], peut diminuer le nombre d'unités candidats. Mais ceci reste à tester, car l'exécution de ces filtres consomme aussi du temps ;

- l'application d'une recherche forward-bakward, double le temps de la deuxième sélection, mais il reste toujours petit par rapport au temps total. Donc il est encourageant de maintenir cette recherche combinée. Ce temps dépend principalement du résultat de la sélection contextuelle (nombre des unités candidates résultantes).

Tableau 4.8: Analyse de la consommation temporelle des fonctions du système HQ_TTS

N° du verset	**Temps d'exécution (s)**					
	Total	Transcription et annotation	1ère étape de sélection	2ème étape de sélection		
				Total	RF	RB
1	1.935	0.150	1.070	0.423	0.247	0.175
2	0.891	0.126	0.436	0.112	0.066	0.045
3	2.318	0.149	1.393	0.207	0.120	0.086
4	1.709	0.160	1.112	0.118	0.068	0.049
5	3.301	0.152	1.322	0.392	0.222	0.168
6	2.997	0.193	1.965	0.258	0.174	0.083
7	2.564	0.175	1.453	0.574	0.356	0.217
8	2.630	0.148	1.725	0.379	0.219	0.159

4.5 Conclusion

Dans ce chapitre, nous avons vu le processus d'évaluation du système HQ_TTS. Les résultats obtenus ont été satisfaisants et encourageants. Le système atteint un taux d'intelligibilité de parole de 91.38 % et un MOS = 3.66 de naturel. De plus, nos approches proposées d'intégrer les SE dans la sélection d'unité et la double recherche forward-backward ont montré aussi leurs bonnes performances par rapport aux autres techniques. Le système a aussi synthétisé une bonne et correcte récitation du Saint Coran.

CONCLUSIONS GÉNÉRALES ET PERSPECTIVES

Le but principal de cette thèse consiste à élaborer un système de lecture automatique du Saint Coran, afin d'aider à sa bonne récitation et l'enseignement des règles de *tajweed*. Ceci a abouti au développement d'un système de synthèse de la parole pour optimiser l'espace mémoire occupé et donne la liberté aux utilisateurs de choisir la partie qu'ils souhaitent écouter.

Afin d'arriver à la meilleure qualité de la parole (intelligible et naturelle), nous avons adopté la Synthèse par Sélection d'Unités (SSU) acoustiques. Cette méthode se base sur la concaténation des unités naturelles bien sélectionnées à partir d'une Base de Données (BD). Par conséquent, sa performance dépend de la richesse de cette BD et l'efficacité de l'algorithme de sélection. Pour cela, nous avons construit notre base personnelle constituée de 11112 unités (diphones et polyphones). La Transcription Orthographique Phonétique (TOP) est une étape essentielle dans ce système, où nous avons basé sur les règles de la lecture en Arabe et de *tajweed* pour accomplir cette phase. Notre contribution principale consiste à améliorer le processus de sélection des unités. Pour cette raison, ce dernier est divisé en deux parties successives et une recherche forward-bakward a été appliquée. De plus, nous avons proposé une nouvelle approche pour la pondération des caractéristiques d'unité utilisées. Pour cette tâche, nous avons développé un Système Expert (SE) qui emploie les spécificités phonétiques et phonologiques de l'Arabe et du Coran.

L'évaluation du système conçu (noté HQ_TTS) a été effectuée par module et globalement, et par l'application des tests objectifs et subjectifs. D'abord, l'évaluation du module de la TOP a donné des résultats identiques à une transcription manuelle. Nos initiatives de diviser l'algorithme de sélection et la recherche forward-bakward ont montré leurs utilités dans le système HQ_TTS, par son optimalité de sélection et son temps réduit par rapport aux approches standard. De plus, Le SE développé a prouvé son efficacité d'améliorer la sélection par rapport à d'autres techniques comme les AGai et AG. L'évaluation de la qualité de la parole synthétisée a donné un excellent résultat en termes d'intelligibilité de 91.38 % et une parole naturelle de 73.2 %. La récitation correcte du SC est achevée par la bonne application des règles de *tajweed*. Le système HQ_TTS a été satisfaisant et a reçu un grand encouragement de la part des participants des tests. Cependant, certains parmi eux ont été ennuyés par la discontinuité de la parole synthétique dans quelques exemples. Ceci, est un problème inévitable dans les systèmes de synthèse par concaténation. En plus, la qualité des sons de la BD a un effet sur ça. Pour surmonter ces problèmes nous

proposons l'extension de la BD par des sons bien enregistrés et plus d'exemples pour les sons rares. Aussi, nous pouvons appliquer les techniques de lissage et de traitement du signal juste pour les parties mal synthétisées pour ne pas modifier tout le signal où nous risquons de le dégrader. L'application des dernières techniques hybrides de synthèse peut optimiser l'espace mémoire, car elles basent sur des modèles statistiques des segments des sons. L'utilisation de ce système n'est pas limitée à la récitation du SC. Le HQ_TTS peut être adapté pour des textes arabes normaux par l'utilisation de la base convenable et un petit réglage dans la TOP, c'est-à-dire en utilisant le même algorithme de sélection.

Nos perspectives pour le système HQ_TTS concernent:

- l'enrichissement par d'autres récitateurs ou types de récitations pour le rendre plus intéressant et utilisable par n'importe quel musulman dans le monde. Ceci peut consommer beaucoup d'espaces mémoires, pour cela, nous pensons de continuer avec les dernières recherches de synthèse comme les wavenet et le monde d'embadding ;

- le HQ_TTS est élaboré sous MATLAB qui est un choix pour sa facilité d'emploi, et les divers Toolbox qu'il propose. Néanmoins, son utilisation nécessite l'installation du MATLAB Runtime qui permet de crée la version exécutable de l'application proposée, mais en même temps le runtime occupe un espace mémoire considérable d'un part et d'autre part MATLAB n'est pas gratuit. Pour cela, une conversion de nos codes vers d'autres langages plus adaptés (comme le JavaScript, C++, C#, etc) et plus accessible est souhaitable pour l'utilisation commerciale. Actuellement, le système utilise 3.85 % de l'enregistrement total du Coran. À cause de cet espace mémoire réduit et son temps de calcul optimisé, nous pouvons l'intégrer sur de petits appareils comme les téléphones portables;

- Le HQ_TTS offre la possibilité de lire n'importe quelle partie du Coran. Par conséquent, nous pensons à intégrer un identificateur de l'arrêt correct (*el-waqf*). Dans le cas où l'utilisateur introduirait une phrase où le waqf n'est pas permis, le système doit proposer de la changer en indiquant la cause. Nous voulons aussi compléter le HQ_TTS par un système de reconnaissance de la parole, pour arriver à un système complet de récitation et d'apprentissage du Sait Coran et ses règles de *tajweed*.

RÉFÉRENCES BIBLIOGRAPHIQUES

[1] ROUIBIA, Soufiane.*Prise en compte de critères acoustiques pour la synthèse de la parole*. Thèse de doctorat : Traitement du signal et Télécommunications. France : Ecole Nationale Superieure des Telecommnunications de Bretagne, 2006,132p.

[2] CHENTIR, Amina.*Etude de la microprosodie en vue de la synthèse de la parole en Arabe standard*. Thèse de doctorat : Electronique. Alger: Ecole Nationale Polytechnique, 2009,109p.

[3] EN-NAJJARY, Taoufik. *Conversion de voix pour la synthµese de la parole*.Thèse de doctorat : Traitement du signal et Télécommunications.France : Université de Rennes I, 2005, 141p.

[4] BENSELAMA, Zoubir-Abdeslem. *Pathologie du Langage Parlé Arabe Cas des Sigmatismes Occlusifs et Constrictifs*. Thèse de doctorat : Electronique. Alger: Ecole Nationale Polytechnique, 2007, 166p.

[5] XAVIER, Florent. *Synthèse vocale integration du français au systeme mary text-to-speech*. Mémoire de mastère. France :Telecom Paristech, 2011, 85p.

[6] AHMED,Ragheb A. فونولوجيا القرآن: دراسة لأحكام التجويد في ضوء علم الأصوات الحديث .Mémoire de magister : Linguistique. Egypte : Université de Ain Chems,2004, 362p.

[7] *Larousse Dictionnaires*. [Consulté le 12/02/2020]. Disponible sur : <http://www.larousse.fr/dictionnaires/francais/ton/78363>

[8] DUTOIT, Thierry."*Introduction au traitement automatique de la parole*. France : Faculté Polytechnique de Mons, notes de cours / DEC2, première édition, 2000.

[9] ELMOUZNI,G. *Introduction au traitement automatique de la parole*.EISTI, cours & TPs , 2010-2011.

[10] AISSIOU, Mohamed.*Application des algorithmes génétiques au décodage acoustico-phonétique de la parole en arabe standard*. Thèse de doctorat : Electronique. Alger: Ecole Nationale Polytechnique, 2008, 156p.

[11] *Lieux d'articulation. riyat warch ann Naafia min tariq el azraq*. [Consulté le 12/10/2020]. Disponible sur : <http://www.riyayatwarch.wordpress.com>

[12] CALLIOPE. *La parole et son traitement automatique.*: Elsevier Masson, 1989.

[13] OUNNAS, Amine. *Synthese de la parole en arabe standard*. Mémoire de magister : Electronique. Alger: Ecole Nationale Polytechnique, 2011,92p.

[14] BALOUL, Sofiane. *Développement d'un système automatique de synthèse de la parole à partir du texte arabe standard voyellé*. Thèse de doctorat, Univerité de Main. Le Mans, France, 2003.

[15] GOUMA, Taoufik *L'emphase en arabe marocain vers une analyse autosegmentale*. Thèse de Doctorat : Linguistique théorique et descriptive. France : Université Paris 8,2013, 161p.

[16] TAYLOR, Paul. *Text-to-Speech Synthesis*. Cambridge: Cambridge University Press, 2009, 627p.

[17] LEMMETTY, Sami. *Review of speech synthesis technology*, Thèse de master : Electrical and Communications Engineering. Finlande : Helsinki University of Technology, 1999, 113p.

[18] LAPRIE, Yves. *Analyse spectrale de la parole*. 2009.

[19] *Section 9.3 feature extraction MFCC vectors*.[Consulté le 14/10/2016]. Disponible sur : <https://www.coursehero.com/file/25382908/mfccpdf>

[20] PRAHALLAD, Kishore. *Speech Technology: A practical inroduction spectrogram, cepstrum and mel frequency analysis*, cours, Internatuinal institue of information technology , Carnegie Mellon University, Hyderbad.

[21] DEMRI, Lyes. *Contribution à l'élaboration d'un système de synthèse par concaténation de la parole expressive*. Thèse de Doctorat : Télécommunication et Traitement de l'Information. Alger : Université des Sciences et de la Technologie Houari Boumediene, 2016, 160p.

[22] Campbell, Nick. Evaluation of Speech Synthesis. In : *Evaluation of Text and Speech Systems*, L. Dybkjær, H. Hemsen, and W. Minker, Eds., ed Dordrecht: Springer Netherlands, 2007, pp. 29-64.

[23] MOURTAN ASSAF, Maria. *A Prototype of an Arabic diphone speech synthesizer in festival*.Thèse de master. Suède : Uppsalla University, 2004.

[24] D'ALESSANDRO, C,et RICHARD,G.*Synthèse de la parole à partir du texte*. Technique de l'ingénieur, 2013.

[25] GUNNEC,David. *Étude des algorithmes sélection d'unités pour la synthèse de la parole à partir du texte*. Thèse de doctorat : Informatique. France :Université de Rennes 1, 2016, 200.

[26] SAGISAKA, Yoshinori. Speech synthesis by rule using an optimal selection of non-uniform synthesis units. In: *International Conference on Acoustics, Speech, and Signal Processing*, 1988, pp.679-682.

[27] HIROKAWA, Tomohisa. "Speech synthesis using a waveform dictionary. In: *First European Conference on Speech Communication and Technology*, 1989, pp. 1140-1143.

[28] HUNT,A.J, et BLACK, Alan.W. Unit selection in a concatenative speech synthesis system using a large speech database. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Atlanta, GA, USA, 1996, pp. 373-376.

[29] DUTOIT, Thierry. Corpus-Based Speech Synthesis. In : *Springer Handbook of Speech Processing*. J. Benesty, M. M. Sondhi, and Y. A. Huang, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 437-456.

[30] Robin. *What is Corpus?*.[Consulté le 20/08/2020]. Disponible sur : < http://language.worldofcomputing.net/linguistics/introduction/ what-is-corpus.html>

[31] ALÍAS,Francesc, FORMIGA Lluís et LLORA, Xavier. Efficient and eliable perceptual weight tuning for unit-selection text-to-speech synthesis based on active interactive genetic algorithms: A proof-of-concept. *Speech Communication*. 2011, vol.53, pp.786-800.

[32] BLACK,Alan.W, et CAMPBELL, Nick. Optimising selection of units from speech databases for concatenative synthesis. In: *European Conference on Speech Communication and Technology*. Madrid, Espagne, 1995, pp. 581-584.

[33] PARK, Seung Seop, KIM, Chong Kyu et KIM, Nam Soo. Discriminative weight training for unit-selection based speech synthesis. In: *Eighth European Conference on Speech Communication and Technology*. Genève, Suisse, 2003, pp.281–284.

[34] KIM, Nam Soo et PARK, Seung Seop. Discriminative training for concatenative speech synthesis.*IEEE Signal Processing Letters*.2004, vol.11, pp.40-43.

[35] ALÍAS,Francesc et LLORA, Xavier. Evolutionary weight tuning based on diphone pairs for unit selection speech synthesis. In : *INTERSPEECH*. Genève, Suisse, 2003.

[36] STROM, Volker. et KING, Simon. Investigating Festival's target cost function using perceptual experiments. In : *INTERSPEECH*, Brisbane, Australie, 2008, pp.1873-1876.

[37] MENG,Helen, KEUNG, Chi Kin, SIU Kai Chung, *et al*.CU Vocal: Corpus-based syllable concatenation for Chinese speech synthesis across domains and dialects. In: *International Conference on Spoken Language Processing*. Denver, USA,2002, pp.2373–2376.

[38] TODA ,Tomoki . *High-quality and flexible speech synthesis with segment selection and voice conversion*.Thèse de doctorat : Traitement de l'information. Japan : Nara Institute of Science and Technology, 2003, 133p.

[39] CAMPILLO, Francisco Díaz, et RODRIGUEZ, Banga.A method for combining intonation modelling and speech unit selection in corpus-based speech synthesis systems. *Speech Communication*.(2006), vol. 48, pp. 941-956.

[40] CLARK,Robert A.J, RICHMOND,Korin et KING, Simon. Multisyn: Open-domain unit selection for the Festival speech synthesis system. *Speech Communication*.2007, vol.49,n°4, pp. 317-330.

[41] PENG, Hu, ZHAO,Yong et CHU, Min. Perpetually optimizing the cost function for unit selection in a TTS system with one single run of MOS evaluation. In: *Internatinal Conference on Spoken Language Processing*. Denver, USA, 2002, pp.1341–1344.

[42] CHU, Min et PENG, Hu. An objective measure for estimating MOS of synthesized speech. In: *European Conference on Speech Communication and Technology*. Aalborg, Denmark, 2001, pp.2087–2090.

[43] DURANT, Eric A, WAKEFIELD, Gregory H, TASELL, Dianne J. Van *et al.* Efficient perceptual tuning of hearing aids with genetic algorithms, *IEEE Transactions on Speech and Audio Processing*.2004, vol.12,n°2, pp.144-155.

[44] ALÍAS,Francesc, LLORA, Xavier,SANZ, Ignasi Iriondo, *et al.* Perception-guided and phonetic clustering weight tuning based on diphone pairs for unit selection

TTS. In : *International Conference on Spoken Language Processing*, Ile de Jeju , Corée de Sude, 2004,pp.1221–1224.

[45] LLORÀ, Xavier, K. SASTRY, D. E. GOLDBERG, A. GUPTA, and L. Lakshmi, "Combating user fatigue in iGAs: partial ordering, support vector machines, and synthetic fitness," presented at the annual conference on Genetic and evolutionary computation, Washington DC, USA, 2005.

[46] CONKIE, Alistair D et ISARD, Stephen. Optimal Coupling of Diphones. In Progress in Speech Synthesis, J. P. H. van Santen, J. P. Olive, R. W. Sproat, and J. Hirschberg, Eds., ed New York, NY: Springer New York, 1997, pp. 293-304.

[47] VEPA, Jithendra *Joint Cost for Unit Selection Speech Synthesis*. Thèse de doctorat Ph.D thesis. Royaume uni : Edinburgh university, 2004,241p.

[48] PFITZINGER, Hartmut.DFW-based spectral smoothing for concatenative speech synthesis. In : *International Conference on Spoken Language Processing*, Ile de Jeju , Corée de Sude, 2004.

[49] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "Tandem-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, F0, and aperiodicity estimation," IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3933-3936, 2008.

[50] KRISHNAMOORTHY,CS et RAJEEV, S. *Artificial intelligence and expert systems for engineers*. vol. 11: CRC press, 1996.

[51] SILER, William et BUCKLEY, James J. *Fuzzy expert systems and fuzzy reasoning*.: Wiley Online Library, 2005.

[52] TEBBI, Hanane. *Modélisation de la synthèse vocale par un système expert*. Thèse de Doctorat : Informatique. Algerie: Université des Sciences et de la Technologie Houari Boumediene, 2019.

[53] CHARBONNIER,Marion.*Les Systèmes Experts : Etat de l'art et application possible aux SIG*. Projet Bibliographique de Mastère. France :École Nationale des Sciences Géographiques, 2008.

[54] TRIPATHI,K P .A review on knowledge-based expert system: concept and architecture. *IJCA Special Issue on Artificial Intelligence Techniques - Novel Approaches & Practical Applications*,2011, pp. 19-23.

[55] HOLLAND,John Henry.*Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*.: MIT press, 1992.

[56] DURAND, Nicolas. *Algorithmes Génétiques et autres méthodes d'optimisation appliqués à la gestion de trafic aérien*. Thèse de doctorat, France : Institut National Polytechnique de Toulouse, 2004.

[57] PELLERIN, E. *Méta-apprentissage des algorithmes génétiques*. Mémoire de Maîtrise : Mathématiques Informatique.Canada: Université du Québec à trois-rivières, 2005.

[58] SIVANANDAM, SN and DEEPA,SN . Genetic algorithms. In : *Introduction to genetic algorithms*.: Springer, 2008, pp. 15-37.

[59] ALSHARIF,Bana, TAHBOUB, Radwan et ARAFEH, Labib. Arabic Text To Speech Synthesis Using Quran Based Natural Language Processing Module. *Journal of Theoretical and Applied Information Technology*.2016, vol.83,n°1 p

[60] SWEED, Ahmed Rouchdi. *the quran teacher*.[Consulté le 12/02/20]. Disponible sur : < https://ar.beta.islamway.net/collection/11899/%D8%A7%D9%84%D9%85%D8%B5%D8%AD%D9%81-%D8%A7%D9%84%D9%85%D8%B9%D9%84%D9%85>

[61] BETTAYEB, Nadjla, Guerti, Mhania. Standard Arabic Talking Clock Based on Diphone Synthesis. In: *the First International Conference on Automatic control, Telecommunications and Signals*. Annaba , Algerie, 2015.

[62] WOJCICKI, K.*HTK MFCC MATLAB. MATLAB Central File Exchange*.[Consulté le 12/10/2020]. Disponible sur : <https://www.mathworks.com/matlabcentral/fileexchange/32849-htk-mfcc-matlab>

[63] EL-IMAM,Yousif. Phonetization of Arabic: rules and algorithms. *Computer Speech & Language*.2004, vol. 18,n°4; pp. 339-373.

[64] BETTAYEB, Nadjla, Guerti, Mhania.A Study to Build a Holy Quran Text-To-Speech System. *International Journal on Islamic Applications in Computer Science And Technology*.2019, vol. 7, n°4,pp. 1-10.

[65] ELSHAFEI, Moustafa, Al-MUHTASEB, Husni, et AL-GHAMDI, Mansour .Techniques for high quality Arabic speech synthesis. *Information Sciences*.2002, vol. 140,n°3-4, pp.255-267.

[66] ALGHAMDI, Mansour, ELSHAFEI, Moustafa et AL-MUHTASEB, Husni. Arabic broadcast news transcription system. *International Journal of Speech Technology*.2007, vol. 10,n°4, pp. 183-195.

[67] FANALS,Lluís Formiga et ALÍAS, Francesc.Perceptual optimization of unit-selection text-to-speech synthesis systems by means of active interactive genetic algorithms. In: *VII Jornadas en Tecnología del Habla and III Iberian SLTech Workshop*, Madrid, Espagne, 2012, pp. 500-509.

[68] XIA,Xianjun, LING,Zhen-Hua , JIANG, Yuan et DAILi-Rong. HMM-based Unit selection speech synthesis using log likelihood ratios derived from perceptual data. *Speech Communication*, 2014, vol. 63-64, pp. 27-37

[69] BETTAYEB, Nadjla, Guerti, Mhania.Speech Synthesis System for the Holy Quran Recitation. *The International Arab Journal of Information Technology*, 2021, vol.18,n°1, pp.8-15.

[70] BETTAYEB, Nadjla, Guerti, Mhania et Ramzan, Naim. A Forward-Backward Dynamic Programming Search For Arabic Unit Selection Speech Synthesis. In : First International Conference on Embedded and Distributed Systems (EDiS), Oran, Algerie, 2017, pp.73-77.

[71] TAYLOR, Paul.Synthesis by concatenation and signal-processing modification.In *Text-to-Speech Synthesis*, ed Cambridge: Cambridge University Press, 2009, pp.412-434.

[72] IMEDJDOUBEN Fayçal et AMRANE Houacine, A. Automatic phonetization of Arabic text. In *Modeling Approaches and Algorithms for Advanced Computer Applications*, A. Amine, A. M. Otmane, and L. Bellatreche, Eds., ed Cham: Springer International Publishing, 2013, pp.85-94.

[73] RARMSAY, Allan, I. ALSHARHAN, Iman et AHMED, Hanady. Generation of a phonetic transcription for modern standard Arabic: A knowledge-based model. *Computer Speech and Language*. 2014, vol. 28,n°4, pp.959-978.

[74] CLARK,Robert A.J, RICHMOND,Korin et KING, Simon. Multisyn: Open-domain unit selection for the Festival speech synthesis system. *Speech Communication*.2007, vol.49,n°4, pp. 317-330.

[75] TAYLOR, Paul.Synthesis techniques based on vocal-tract models. In *Text-to-Speech Synthesis*, ed Cambridge: Cambridge University Press, 2009, pp. 387-411.

[76] BETTAYEB, Nadjla. *Comparaison GA*.[Consulté le 09/09/2020]. Disponible sur : <https://docs.google.com/forms/d/19Gq4Z1ZVxJODj2wPUq3IVJANbvG9dAEwjVFnXW0BKb8/edit>

[77] BETTAYEB, Nadjla. *ES_comparaison_EN*.[Consulté le 09/09/2020]. Disponible sur : < https://docs.google.com/forms/d/1gxa-KTQnIhph2rFFSExFyA3Y3ddGWhbEN2567D4RQ-0/edit>

[78] BETTAYEB, Nadjla. *ES_comparaison_AR*.[Consulté le 09/09/2020]. Disponible sur : <https://docs.google.com/forms/d/1XduLPrIoEeeAj9zGJTeF67uj40faHy12dhw6ySpOXIE/edit?usp=drive_web>

[79] BETTAYEB, Nadjla. *TTS evaluation_EN*.[Consulté le 09/09/2020]. Disponible sur : <https://docs.google.com/forms/d/17wxzNsoBpZwzh1ZG0guTee3ZML-sceH7WIZWHu-KHVE/edit>

[80] BETTAYEB, Nadjla. *TTS evaluation_AR*.[Consulté le 09/09/2020]. Disponible sur : <https://docs.google.com/forms/d/1zMnQV2CiLvT5zOjYCKHIcfqOycEyYBr4CFN-ZDI6gbw/edit?usp=drive_web>