View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Institut National Polytechnique de Toulouse (Theses)

Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Signal, Image, Acoustique et Optimisation

Présentée et soutenue par :
Lionel KOENIG

le : vendredi 28 janvier 2011

Titre :

Masquage de pertes de paquets en voix sur IP

Ecole doctorale :
Mathématiques Informatique Télécommunications (MITT)

Unité de recherche :
Institut de Recherche en Informatique de Toulouse (IRIT)

Directeur(s) de Thèse :
Corinne Mailhes
Régine André-Obrecht

Rapporteurs :

Régine Le Bouquin-Jeannès
Denis Jouvet

Membre(s) du jury :
Laurent Girin, Professeur des universités à l'INP Grenoble, Président
Denis Jouvet, Directeur de Recherche à l'INRIA Nancy, Rapporteur
Régine Le Bouquin-Jeannès, Professeur de l'Université de Rennes 1, Rappoteur
Corinne Mailhes, Professeur des universités à l'INP Toulouse, Directrice de thèse
Régine André-Obrecht, Professeur de l'Université de Toulouse 3, co-Directrice de thèse
Niels Nielsen, Ingénieur à la société Intel Corporation à Toulouse, Examinateur

Table des matières

Remerciements

Ce travail est le fruit d'une collaboration entre la société Freescale Semiconducteur qui m'a employé et l'Institut de Recherche en Informatique de Toulouse.

Tout d'abord je tiens à remercier les rapporteurs de mon travail qui ont, à partir d'un document plutôt succinct, fait des remarques pertinentes dont j'espère avoir tenu compte dans la version finale de ce document. Je remercie les membres du jury pour leurs remarques ou questions qui permettent d'approfondir ma réflexion.

Mes directrices de thèses, Régine Andre-Obrecht et Corinne Mailhes, ainsi que les responsables Freescale, Serge Fabre, Robert Krutsh, Niels Nielsen et Ioanita Mircea.

Une mention spéciale pour Elsa, mon amie, sans qui je ne serai pas arrivé jusqu'ici. Elle témoignera que les années de thèses, de surcroît combinées avec une formation de musicienne intervenante brillamment réussie, ne sont pas toujours roses. Merci.

Merci aux courageuses chasseuses de fautes de français : Marie, Ghislaine, encore Corinne, à nouveau Régine, Sylvie.

Merci a mes collègue de l'IRIT, Hervé, Hélène, Maxime, Julien Jérôme, Ioannis, Reda, Christine, Patrick, Philippe, pour leurs bonnes et mauvaises humeurs.

Un grand merci également à ma famille qui m'a soutenu tout au long de

cette aventure, a réalisé le pot.

Merci à l'ex-équipe audio de Freescale, Sylvain, Hans, Hervé, Céline, Aurélien, Cédric, Guillaume. J'ai également une petite pensée pour Sylvette du service des ressources humaines.

Mes amis, musiciens ou non, ont également indirectement contribué à cette aventure. Je pense particulièrement à Adeline, le CA de l'EIA, Gilles.

Notations

Symbole	Description
t	Le temps continu. Abusivement le temps discret.
N	Nombre d'échantillons dans une trame d'analyse.
$x(n)$	n^{e} échantillon.
f_s	Fréquence d'échantillonnage.
$\tilde{f}$	Fréquence normalisée. $\tilde{f} = \frac{f}{f_s}$ où f est la fréquence en Hz.
P	Probabilité.
τ	Instant de la première trame perdue.
L	Nombre de trames perdues.
J	Nombre de trames futures disponibles.
D	Dimension du vecteur d'observation.
$\boldsymbol{\phi}_t$	Vecteur d'observation de la trame audio d'indice t.
$\boldsymbol{\phi}_{t_1}^{t_2}$	séquence des vecteurs d'observation $\boldsymbol{\phi}_{t_1}, \cdots, \boldsymbol{\phi}_{t_2}$.
$\boldsymbol{\psi}_t$	Vecteur associé à la génération de la trame audio d'indice t.
Q	Nombre d'états du modèle de Markov caché.

q_t Variable aléatoire représentant le numéro de l'état à l'instant t.

$\boldsymbol{A} = (a_{i,j})_{i,j \in [\![1,Q]\!]^2}$ Matrice de transition du modèle de Markov caché.

$a_{i,j} = \mathrm{P}\left(q_t = j | q_{t-1} = i\right)$ La probabilité de passer de l'état i à l'état j.

b_i Densité de probabilité d'observation de l'état i.

$$b_i(\boldsymbol{\phi}_t) = \mathrm{P}\left(\boldsymbol{\phi}_t | q_t = i\right).$$

$\alpha_t(i)$ Probabilité d'avoir la séquence d'observations $\boldsymbol{\phi}_1^t$ et d'être dans l'état i à l'instant t.

$$\alpha_t(i) = \mathrm{P}\left(\boldsymbol{\phi}_1^t, q_t = i\right)$$

$\boldsymbol{\alpha}_t$ $= [\alpha_t(1) \cdots \alpha_t(Q)]^\dagger$.

$\beta_t(i)$ Probabilité d'avoir la séquence d'observations $\boldsymbol{\phi}_{t+1}^{\tau+L+J-1}$ sachant que l'on est dans l'état i à l'instant t.

$$\beta_t(i) = \mathrm{P}\left(\boldsymbol{\phi}_{t+1}^{\tau+L+J-1} | q_t = i\right)$$

$\boldsymbol{\beta}_t$ $= [\beta_t(1) \cdots \beta_t(Q)]^\dagger$.

Les vecteurs et matrices sont notés en majuscule et en **gras**. Leurs coordonnées sont notées en minuscule. Ainsi $\boldsymbol{A}$ est une matrice alors que a_{ij} est l'élément de la i^{e} ligne et la j^{e} colonne. Les suites d'éléments u_s pour s variant de i à j sont notées $\{u\}_i^j$ et u_i^j quand la confusion n'est pas possible. $\boldsymbol{A}^\dagger$ est la matrice transposée de $\boldsymbol{A}$. $\det(\boldsymbol{A})$ est le déterminant de la matrice $\boldsymbol{A}$.

On notera $x \sim \mathcal{N}$ le fait que la variable aléatoire x suit la loi $\mathcal{N}$. De plus, la loi normale de moyenne $\boldsymbol{\mu}$ et de matrice de covariance $\boldsymbol{\Sigma}$ sera notée $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

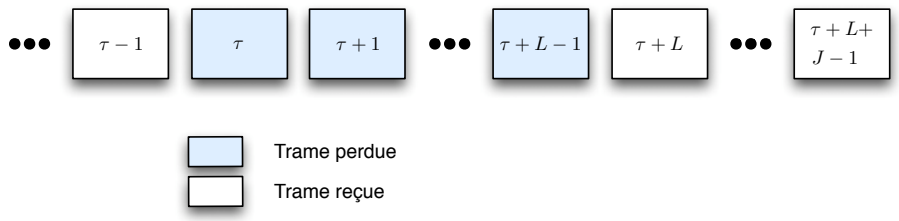


FIGURE 1 – Indices des pertes de paquets.

*L*es L trames (en bleu) à partir de la trame d'indice τ sont manquantes. Les trames avant le temps τ, c'est à dire d'indices $1, \cdots \tau - 1$, ainsi que les J trames $\tau + L, \cdots, \tau + L + J - 1$ sont reçues.

Glossaire

CSLU	Center for Spoken Language Understanding (OGI)
DSP	Denstité Spectrale de Puissance
FER	Frame Erasure Rate
FIR	Finite Impulse Response (cf. RIF)
HMM	Hidden Markov Model
HNM	Harmonic Noise Model
HTK	Hidden markov ToolKit
IIR	Infinite Impulse Response (cf. RII)
IP	Internet Protocol (RFC 791)
ITU	International Telecommunication Union
LIMSI	Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur
LSF	Line Spectrum Frequencies
MFCC	Mel Frequencies Cepstral Coefficients
MOS	Mean Opinion Score
OGI	Oregon Graduate Institute of Technology
PESQ	Perceptual Evaluation of Speech Quality
PLC	Packet Loss Concealment
PSOLA	Pitch Synchronous OverLap and Add
RII	Réponse Impulsionnelle infinie (en anglais IIR)
RIF	Réponse impulsionnelle finie (en anglais FIR)
RFC	Request For Comments

TCP	Transport Control Protocol (RFC 793)
UDP	User Datagram Protocol (RFC 768)

Chapitre 1

Introduction

Dans un monde où le multimédia prend de plus en plus de place dans notre quotidien, la téléphonie occupe une place majeure qu'elle passe par le réseau GSM (télephone portable) ou par internet sur le réseau IP (Internet Protocol). Réduction des coûts, simplicité apparente d'utilisation, enrichissement du contenu, la téléphonie apporte son lot d'améliorations. Elle doit également, pour pouvoir se démocratiser, relever un certain nombre de défis techniques. Parmi ces défis, la gestion des paquets perdus est l'objet de notre étude. En effet, pour être transmis à travers le réseau choisi (internet ou cellulaire), le signal de parole est "découpé" en paquets représentant entre 20 et 30 millisecondes de signal. A la réception, il arrive, pour diverses raisons qui seront précisées par la suite, qu'un ou plusieurs paquets soient manquants. Il faut alors proposer un algorithme de recouvrement de ces paquets afin de ne pas laisser de manques dans le signal de parole reconstruit. Notre travail de thèse est centré sur la création d'un nouvel algorithme de compensation des paquets perdus.

Ce chapitre présente dans un premier temps le principe de fonctionnement d'un service de téléphonie. Dans une seconde partie, nous présentons succinctement le signal de parole, ses propriétés et ses différentes modélisations. La dernière partie situe le cadre de notre étude.

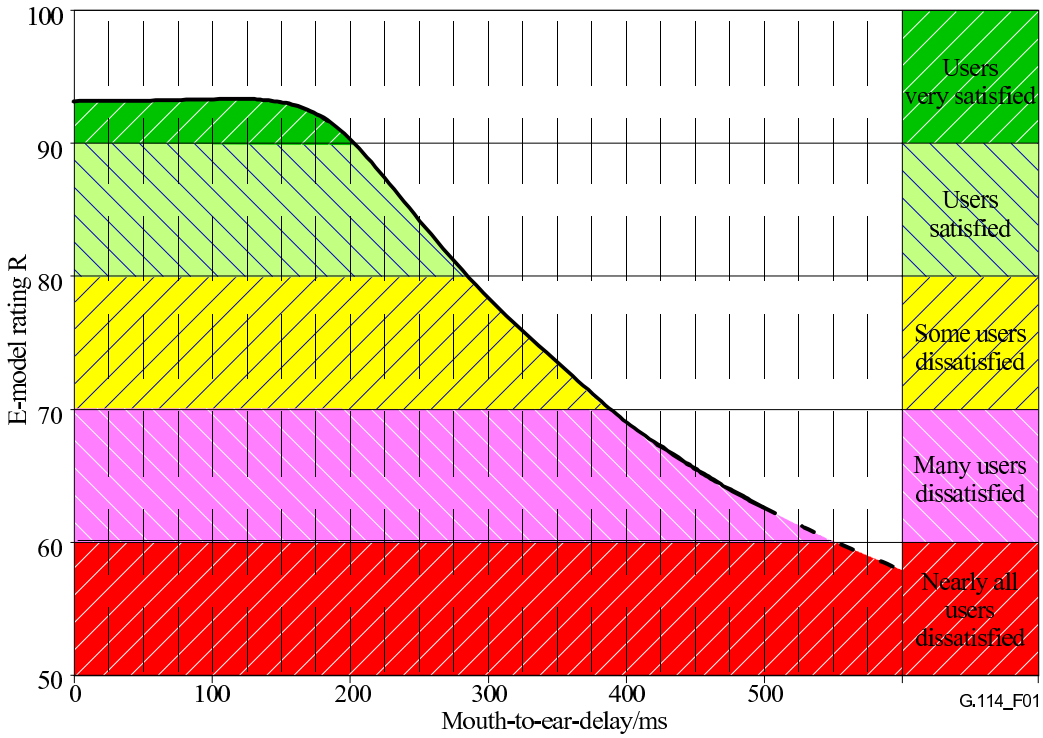


FIGURE 1.1 – Effet du retard sur la qualité de la conversation.
Extrait de [1].

1.1 Système de téléphonie

Le service de téléphonie permet l'échange en **temps réel** de la voix entre deux interlocuteurs distants, assurant ainsi une conversation. Le temps de trajet, c'est-à-dire le temps mis par le son pour aller de la bouche d'un interlocuteur à l'oreille de l'autre, doit être le plus court possible. Pour que la conversation soit efficace, interactive et de bonne qualité, il est nécessaire que ce temps de transmission soit borné. La recommandation G.114 [1] de l'ITU préconise un temps de transmission inférieur à $300ms$ pour avoir une bonne qualité de conversation. La qualité d'un système de téléphonie peut se mesurer à l'aide du E-Model. Ce modèle prend des valeurs entre 0 et 100 en tenant compte à la fois du délai bouche-oreille, de la dégradation introduite par le vocodeur, des effets dûs aux pertes de paquets et de la dégradation engendrée

par les algorithmes acoustiques comme l'annulateur d'écho, ou la suppression du bruit. La courbe de la figure 1.1 extraite de la recommandation G.114 [1] donne les valeurs maximales atteignables par le modèle en fonction du délai bouche-oreille.

La voix sur IP est un service de téléphonie utilisant le réseau internet comme support au transport de la voix. La voix numérisée est transmise sous forme de petits paquets de l'ordre de 10 à $30ms$ sur le réseau internet.

Chaque paquet est compressé à l'aide d'un codeur de parole (G711 [2], G729 [3], AMR, AMR-WB [4], iLBC [5, 6], ...). Puis le paquet compressé est encapsulé dans un paquet de niveau transport. Dans le cas de la voix sur IP, le protocole de transport utilisé est généralement UDP c'est-à-dire un protocole de communication en mode non-connecté : les paquets sont envoyés mais, contrairement à TCP, chaque paquet est indépendant des autres et aucun mécanisme assurant que le paquet est bien arrivé n'est mis en place. Le paquet transport est ensuite lui-même encapsulé dans un paquet IP (couche réseau) pour être transmis sur le réseau internet.

Le paquet est transporté de l'expéditeur (*near end speaker*) au destinataire (*far end speaker*) à travers le réseau de routeurs et d'équipements. De par la conception du réseau internet, tous les paquets n'empruntent pas le même trajet. De ce fait, les temps de parcours ne sont pas identiques pour tous les paquets et l'ordre d'arrivée des paquets diffère de l'ordre d'émission des paquets. Il est alors nécessaire avant de décompresser les paquets de les remettre dans l'ordre. C'est le rôle du tampon de compensation de gigue (voir la figure 1.2). Sur cette figure, le tampon de compensation de gigue permet de redistribuer les paquets reçus dans l'ordre. Il arrive, notamment dans le cas de réseaux IP, que ce tampon contienne des paquets situés temporellement dans le futur par rapport à l'instant de lecture du récepteur.

En voix sur IP, le réseau servant de support au transport est le réseau internet. Celui-ci fonctionne sur le modèle "Best-Effort". Chaque paquet est transporté du mieux possible d'un bout à l'autre du réseau. Néanmoins, chaque paquet n'empruntant pas le même chemin que son prédécesseur,

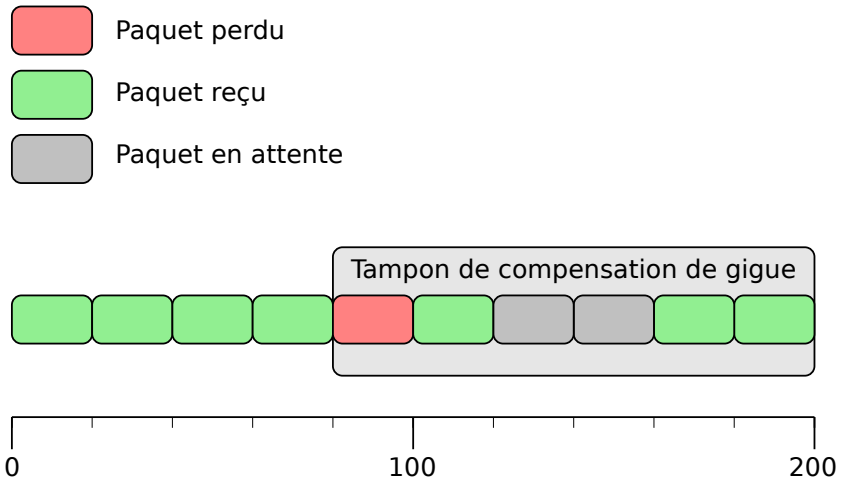


FIGURE 1.2 – Principe du tampon de compensation de gigue.

l'ordre d'arrivée est ainsi bousculé. De plus, des phénomènes de congestion (embouteillage) font que des équipements réseaux (routeurs) peuvent supprimer des paquets pour alléger la charge. On parle alors de « *pertes de paquets* ».

On peut ajouter que le sous-système de reproduction sonore (sur un ordinateur, ce rôle est joué par la carte son) demande périodiquement des échantillons à envoyer au convertisseur numérique - analogique. Ceci oblige le décodeur de parole à produire régulièrement des échantillons en consommant les paquets audios. Or, dans le cas où le paquet qui doit être joué n'est pas encore arrivé, le système, dans l'obligation de transmettre les échantillons sonores, se comporte comme si le paquet était perdu. Dans ce cas précis, il est probable que le tampon de compensation de gigue a déjà reçu les quelques (de l'ordre de trois ou quatre) paquets qui suivent. Ce phénomène provient du fait que les paquets empruntent des routes différentes et ont donc des temps de parcours différents. Ces paquets sont alors situés dans le "*futur*" pour le sous-système de reproduction sonore.

Pour résumer, les causes de pertes de paquets sont multiples :

- Désordre à l'arrivée des paquets (jitter),
- Congestion dans les nœuds du réseau (routeurs),

- Erreurs de transmission,
- Retard à l'arrivée du paquet.

Une solution à ce problème est le masquage de perte de paquets (*Packet Loss Concealment* - PLC en anglais).

1.2 Le signal de parole

Dans le cadre de notre étude, nous nous intéressons à un signal très particulier, le signal de parole. Avant de poser le problème qui fait l'objet de ce travail, il est nécessaire de rappeler les propriétés connues de ce signal qui pourront être utiles à notre étude ainsi que les différents et principaux modèles de ce signal qui ont été proposés dans la littérature.

1.2.1 Propriétés du signal de parole

Le signal de parole est issu de l'enregistrement de la voix à l'aide d'un microphone et résulte de la production de la voix par l'être humain. La diversité des morphologies, des individus, des âges, de la prononciation, etc... font du signal de parole un signal avec une grande variabilité. Néanmoins, le mécanisme de production est le même pour tous les êtres humains.

La génération de l'onde acoustique se fait au niveau des cordes vocales à l'intérieur du larynx (cf. figure 1.3). Deux types de production d'énergie sont possibles :

- *voisée* dans laquelle les muscles du larynx placent les cordes vocales côte à côte. Sous l'influence de l'air expulsé par les poumons, ces cordes vocales vibrent et produisent ainsi une onde quasi-périodique (voir figure 1.4). La période fondamentale de cette onde sonore est appelée *pitch*. Le pitch évolue aux alentours de $120Hz$ pour un homme, $240Hz$ pour une femme et $400Hz$ pour un enfant.

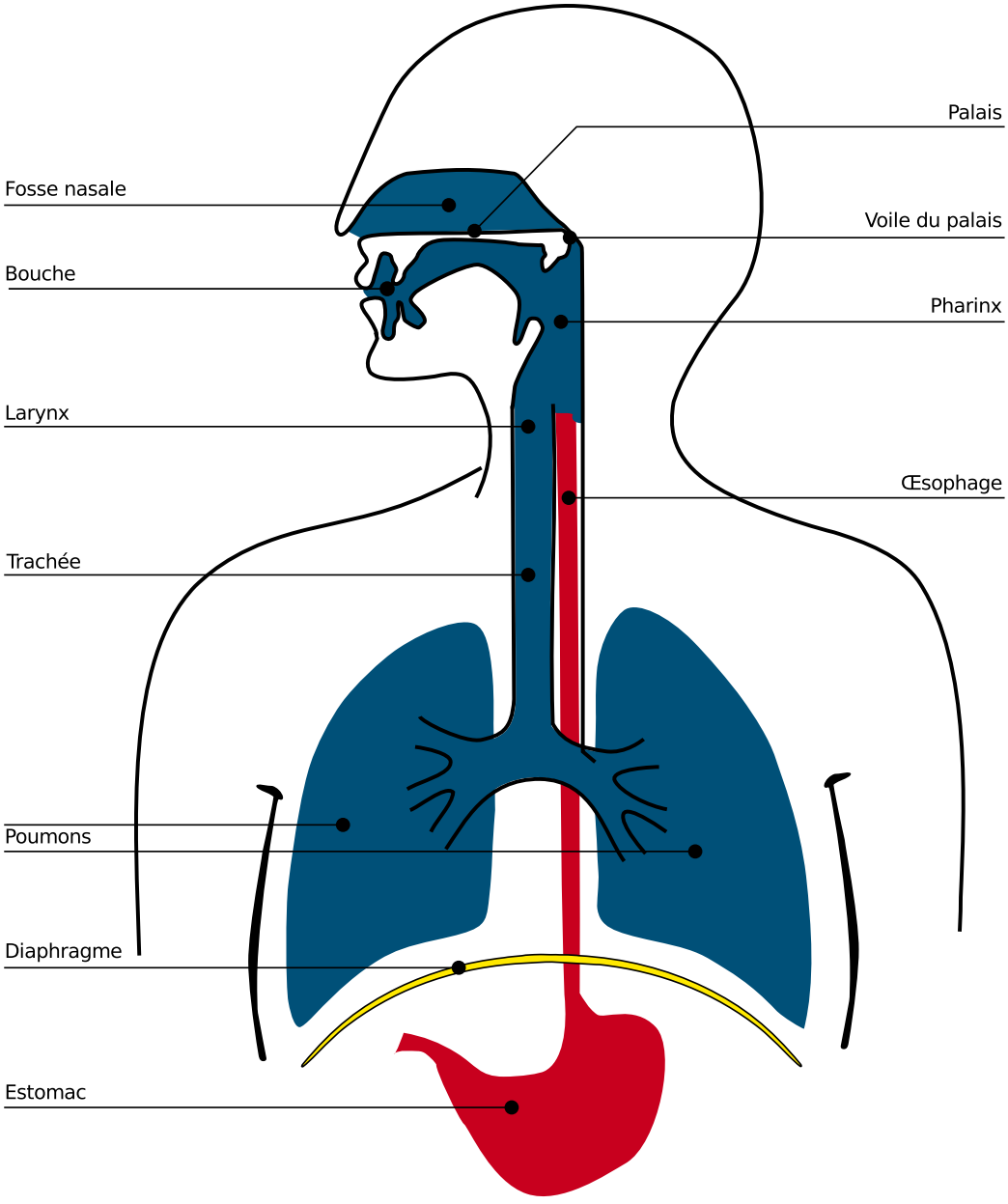


FIGURE 1.3 – Schéma du système de production de la voix.
inspirée de `http://catalogue.ircam.fr/sites/Voix/`

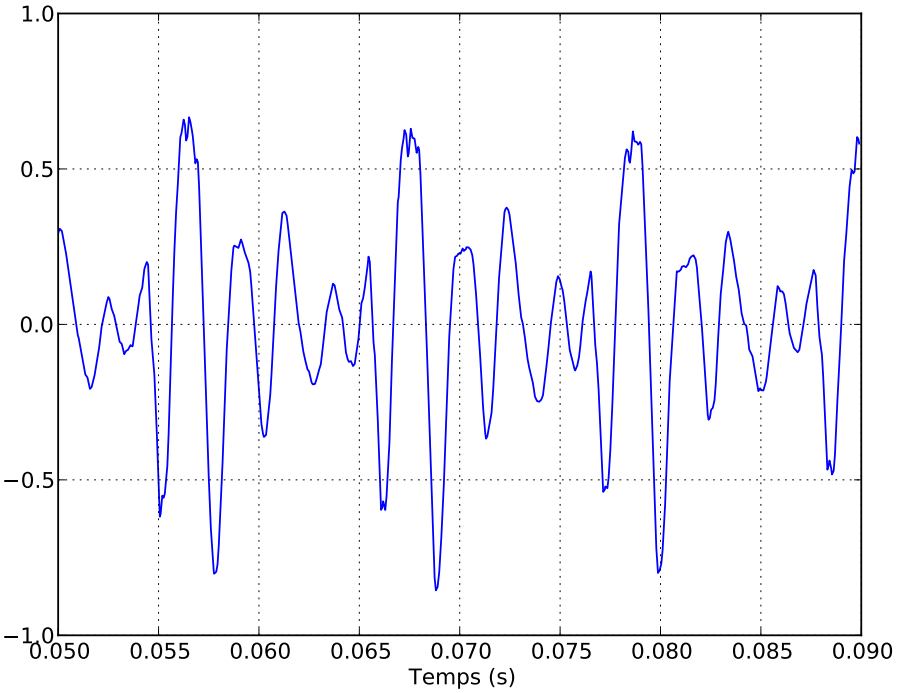


FIGURE 1.4 – 40ms de signal de parole quasi-périodique.

- *non-voisée* dans laquelle l'air passe à travers les cordes vocales sans les faire entrer en vibration. Cet air passe à travers un rétrécissement au niveau de la glotte ce qui entraîne des turbulences. Il en résulte un son apériodique (voir figure 1.5).

Ces deux processus de production de la parole peuvent intervenir simultanément comme par exemple dans le son /z/ présent dans le mot *ro**s**eau* (/ʁɔzɔ/), illustré sur la figure 1.6, son pour lequel il sera difficile de prendre une décision voisée ou non-voisée.

Après avoir franchi le larynx, cette onde acoustique excite différents résonateurs formés par le pharynx et les cavités buccale et nasale pour être finalement diffractée par les lèvres. Les différentes fréquences de résonances de ces résonateurs sont appelées *formants*.

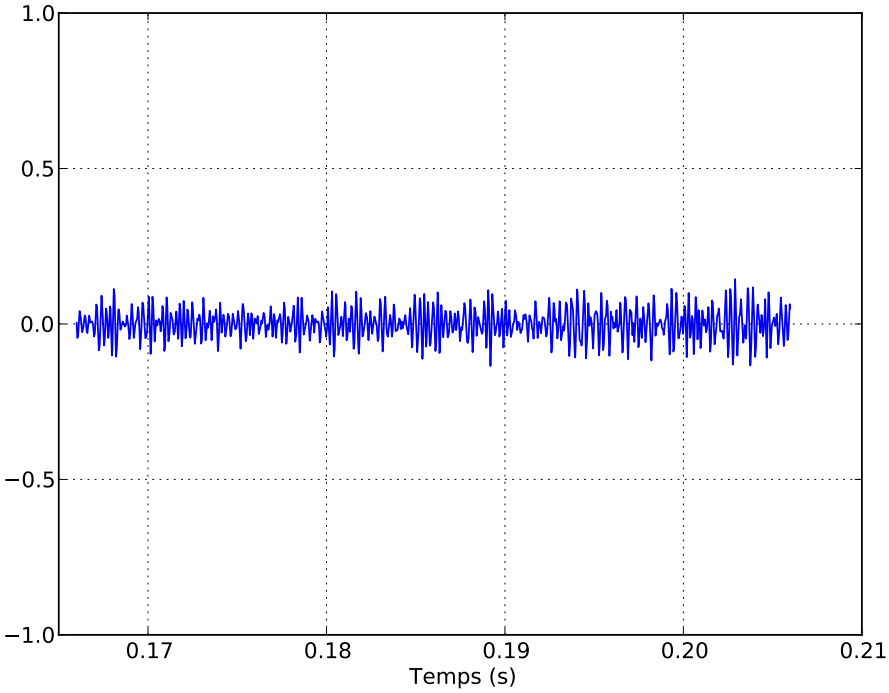


FIGURE 1.5 – 40ms de signal de parole apériodique.

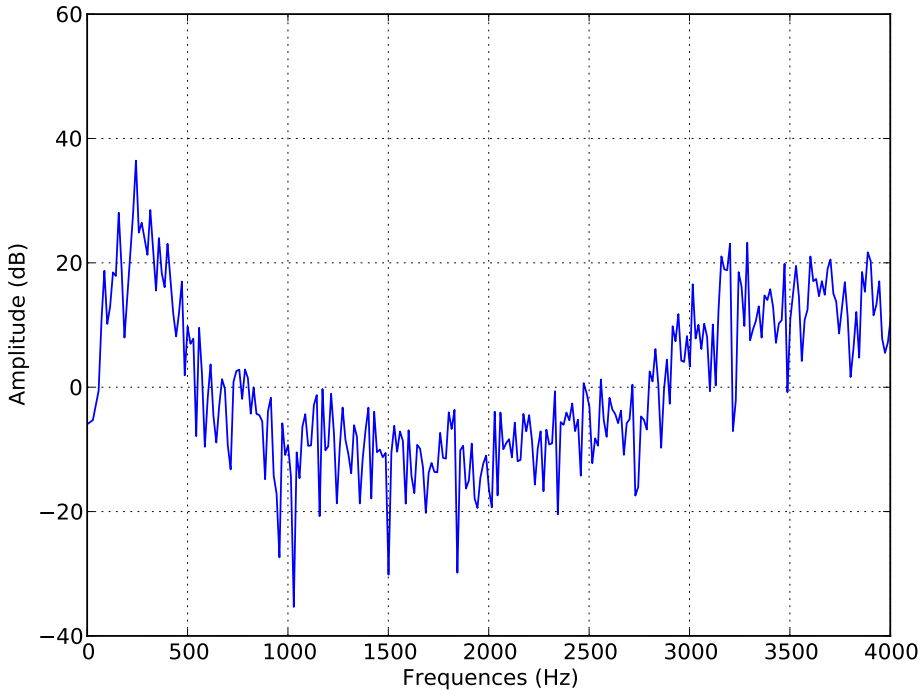


FIGURE 1.6 – Densité spectrale de puissance estimée du son /z/ de *ro**s**eau* (/ʁɔzɔ/).

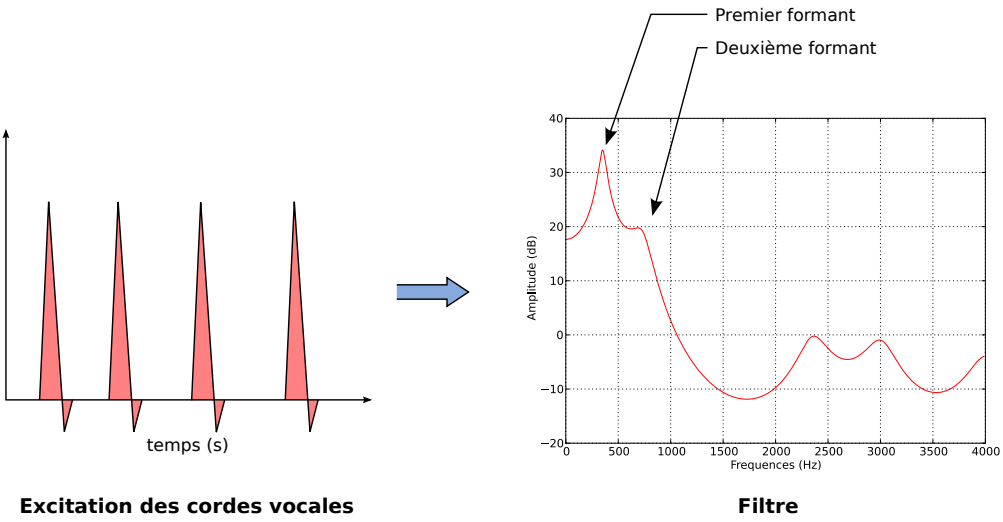


FIGURE 1.7 – Principe du modèle source-filtre.

1.2.2 Différents modèles du signal de parole

Que l'on s'intéresse à la parole pour le codage, la reconnaissance du locuteur, la transcription orthographique, phonétique ou pour la synthèse à partir de texte écrit, l'utilisation de modèles du signal de parole est toujours à envisager. Ainsi, dans le cadre général du traitement de la parole, plusieurs types de modèles ont été proposés dans la littérature. Nous présentons les grandes lignes de ces modèles dans ce qui suit.

Modèle source-filtre De la description physiologique du paragraphe précédent découle une modélisation du signal de parole : le modèle source-filtre. Le principe de ce modèle, résumé sur la figure 1.7, est élémentaire et a servi de base à de nombreux codeurs de parole.

Le signal de parole est modélisé comme la sortie d'un filtre tout-pôle excité par une suite d'impulsions régulières (son-voisé) ou par un bruit blanc (son non-voisé). De nombreuses études ont porté sur la façon de concevoir le filtre ainsi que sur différentes variantes de l'entrée permettant d'améliorer la modélisation en se libérant du choix binaire difficile du son voisé ou non-voisé.

Modèle harmonique plus bruit (HNM) Le signal de parole peut également être modélisé comme un signal périodique (une fréquence fondamentale, le pitch, et toutes ses harmoniques) auquel est adjoint un bruit. Dans ce cas, les paramètres sont :

- la fréquence fondamentale qui dépend du temps,
- le nombre d'harmoniques,
- la phase et l'amplitude de chaque harmonique,
- la répartition spectrale du bruit,
- le ratio entre la partie harmonique et la partie bruitée qui permet ainsi de pondérer l'aspect voisé/non-voisé des signaux modélisés.

Aspects linguistiques La parole est un agencement ordonné de différents sons élémentaires, les *phones*. Ces unités sonores ont un équivalent linguistique, les *phonèmes* qui sont définis par le dictionnaire de l'Académie française[1] de la manière suivante :

> La plus petite unité sonore d'une langue donnée, caractérisée par des traits distinctifs, et qui, combinée à d'autres, sert à former des unités signifiantes telles que les morphèmes, les mots, les phrases, etc.

Ainsi *zona* (en phonétique : /zɔna/) et *sauna* (en phonétique : /sɔna/) sont deux mots qui ne se distinguent que par un phonème. Un autre exemple est *patte* (/pat/) et *pâte* (/pɑt/).

1.3 Cadre de recherche

L'objectif de cette thèse est d'étudier les systèmes de masquage de pertes de paquets. Devant la diversité des codeurs de parole, **nous avons choisi**

[1] Académie française : `http://www.academie-francaise.fr/dictionnaire/`

de nous focaliser sur les systèmes indépendants du vocodeur. Ainsi, cette étude s'appuiera uniquement sur les propriétés du signal de parole et les connaissances acquises dans le domaine du traitement automatique de la parole. Le problème se situe au niveau du récepteur : une fois le signal de parole reconstruit, après décodage, il reste des segments de signal manquants ("trous") correspondant aux paquets perdus. L'estimation du signal de ces segments est appelée *masquage de pertes de paquet*. Nous avons également remarqué que dans certains cas, à cause du phénomène de gigue dans l'arrivée des paquets IP, une petite portion de signal située juste après le "trou" (donc dans le "*futur*" pour le récepteur) peut être disponible. Le signal de parole est alors habituellement échantillonné à $F_e = 8kHz$ et une trame perdue de 10ms correspond à 80 échantillons perdus à approximer.

La suite de ce manuscrit s'attache dans un premier temps à décrire quelques systèmes de références dans le domaine ; ce sera l'objet du chapitre 2 suivant.

Le chapitre 3 présente des jeux de données standardisées que nous allons utiliser tout au long de notre étude pour valider nos résultats. Les corpora de parole ont été choisis pour la reconnaissance nationale et internationale dont ils bénéficient dans la communauté du traitement de la parole.

Le chapitre 4 s'intéresse à un nouveau paramètre que nous avons proposé et qui sera utilisé ensuite dans le système de masquage de pertes de paquets que nous étudierons.

Le chapitre 5 détaille les différents modèles de Markov cachés envisagés qui constituent le cœur de notre système. Il aborde également le problème de la recherche du meilleur chemin dans ces modèles de Markov cachés lorsque des trames de signal sont manquantes, problème pour lequel nous proposons une modification de l'algorithme de Viterbi.

Enfin, le chapitre 6 combine toutes les briques de notre système et en présente les résultats obtenus sur les corpora de parole, ce qui nous conduit aux conclusions et perspectives du chapitre 7.

Chapitre 2

Panorama des méthodes de masquage de pertes de paquets

Le masquage de pertes de paquets est un domaine de recherche avec beaucoup d'implications industrielles. Ce chapitre s'efforce de dresser un panorama des principales techniques et méthodes existantes servant à masquer les pertes de paquets (Packet Loss Concealment, PLC).

Deux grandes stratégies cohabitent dans ce domaine : elles sont basées soit sur l'émetteur des paquets, soit sur le récepteur. Les méthodes reposant sur l'émetteur s'appuient sur l'ajout de redondances et de codes correcteurs d'erreurs. Une méthode simple et naïve consiste à envoyer deux flux de données, le premier à fort débit et haute qualité de description et le second avec une description sommaire du signal mais un débit beaucoup plus réduit [7, 8, 9]. Ces méthodes peuvent être perfectionnées et optimisées en terme de bande passante en envoyant une description multi-échelle du signal de parole. Le codec SILK [10] utilisé par le logiciel de téléphonie Skype utilise de tels mécanismes.

Les systèmes basés sur le récepteur sont détaillés dans la suite de ce document puisqu'ils définissent le cadre des recherches exposées. Dans ce domaine, de nombreuses méthodes de masquage de pertes de paquets ont été développées. Beaucoup de codeurs de parole modernes intègrent un mécanisme de

masquage des pertes de paquets comme par exemple [11, 12, 13, 3, 5].

Ces mécanismes se divisent en trois grandes catégories :

l'omission : Les paquets perdus sont remplacés par des trames de silence. Cette méthode bien que possédant un coût calculatoire nul produit un résultat de qualité médiocre : dès que la taille du trou dépasse les $20ms$, l'auditeur perçoit une dégradation.

la répétition : Les éléments de signal manquant sont remplacés par les paquets précédents. Il s'agit donc de répéter le ou les derniers paquets reçus. Ce type de masquage rend imperceptible des trous de petite taille.

les modèles de parole : De coût calculatoire plus important, ces méthodes extrapolent ou interpolent les paramètres d'un modèle de parole du signal reçu sur le signal manquant.

2.1 Réplication de formes d'ondes

Ces méthodes de masquage de pertes de paquets reposent sur l'insertion directe d'une forme d'onde dans le trou. La plus simple consiste à répéter le dernier élément reçu. Plusieurs études proposent néanmoins des versions plus élaborées.

2.1.1 G711 Annexe 1

Le système de masquage de pertes de trames audio proposé pour le codeur G711 est certainement le plus utilisé. Il fait l'objet d'une recommandation (norme) dans l'annexe 1 (*A high quality low-complexity algorithm for packet loss concealment with G.711*) de la norme du codeur de parole G711 [2] (*Pulse code modulation (PCM) of voice frequencies*).

Lorsqu'un paquet est reçu, une copie du signal décodé est enregistrée dans un tampon de $48.75ms$ (390 échantillons) pour calculer le pitch. Le résultat (signal de sortie) est retardé de $3.75ms$ (30 échantillons) pour effectuer une superposition / admission (OLA, *overlap and add*) lors de la fin de l'épisode de perte et assurer ainsi un fondu enchaîné entre le son reçu et le son issu du masquage.

Lors de la première trame perdue, la période fondamentale (pitch) du tampon d'historique est détectée soit à l'aide d'une méthode d'autocorrélation soit par l'intermédiaire d'une fonction moyenne des différences (*average mean difference function*) similaire à celle utilisée par le détecteur de fréquence fondamentale YIN [14].

Le signal synthétique est alors généré en se basant sur la $\frac{5}{4}^{\text{e}}$ dernière période du signal. Ces périodes sont alors répétées avec une opération de fenêtrage afin de synthétiser les 10 premières millisecondes de signal manquant.

La génération du signal synthétique après les 10 premières millisecondes est effectuée avec plus de périodes : en effet, si une seule période est utilisée, le signal synthétique présentera un phénomène de sifflement. Ainsi pour synthétiser le signal de la 10^{e} à la 20^{e} milliseconde, deux périodes sont utilisées. Au-delà de la 30^{e} milliseconde, trois périodes sont utilisées.

Une atténuation est appliquée au fur et à mesure que l'effacement se prolonge. Si elle n'est pas appliquée durant les 10 premières millisecondes, elle est linéaire avec un taux de 20% par $10ms$ jusqu'à faire disparaître le signal synthétique au bout de $60ms$.

Adrian Susan et Mihai Neghina [15] étendent cet algorithme afin de supprimer le délai introduit pour la superposition / admission. Ils proposent d'utiliser la prédiction linéaire afin de générer le signal nécessaire au recouvrement. Ainsi le système de masquage n'introduit aucun retard dans la chaîne de traitement audio.

2.1.2 Estimation du signal par interpolation linéaire

Notamment dans le cas où le signal est présent à la fois avant (ce qui est logique) et après le segment perdu, Kondo et al. [16] proposent de faire une interpolation linéaire échantillon après échantillon de la trame précédente avec la suivante. Supposons que chaque trame contienne N échantillons et notons τ le numéro de la trame perdue. Ainsi le n^{e} échantillon $x(n)$ du segment perdu τ est obtenu comme une combinaison linéaire des prédictions linéaires de la trame précédente $\hat{x}^{(f)}(n)$ et de la trame suivante $\hat{x}^{(b)}(n)$:

$$\hat{x}(n) = \frac{n}{N}\hat{x}^{(f)}(n) + \frac{N-n}{N}\hat{x}^{(b)}(n) \tag{2.1}$$

avec

$$\hat{x}^{(f)}(n) = -\sum_{i=1}^{P} a_i^{(f)} \cdot \hat{x}(n-i)$$

$$\hat{x}^{(b)}(n) = -\sum_{i=1}^{P} a_i^{(b)} \cdot \hat{x}(n-i+N+1)\cdot$$

$\left\{a^{(f)}\right\}_{i=1,\cdots,P}$ et $\left\{a^{(b)}\right\}_{i,\cdots,P}$ sont les coefficients auto-régressifs obtenus par une régression de Levinson-Durbin [17] sur respectivement la trame précédente « *forward* » et la trame suivante « *backward* ».

2.2 Utilisation de modèles de parole

Les algorithmes les plus évolués de masquage de pertes de paquets reposent sur des modèles de parole.

L'un des modèles les plus utilisés est le modèle source-filtre comme présenté dans le paragraphe 1.2.2 : un signal d'excitation est généré pour simuler la source (située au niveau du larynx) puis filtré par le système acoustique (larynx / conduits bucal et nasal / lèvres) pour produire un son. Nombre de codeurs de parole [13] sont basés sur ce modèle en utilisant pour excitation un dictionnaire et un filtre numérique tout-pôle d'ordre limité (en pratique d'ordre 10) pour filtre acoustique.

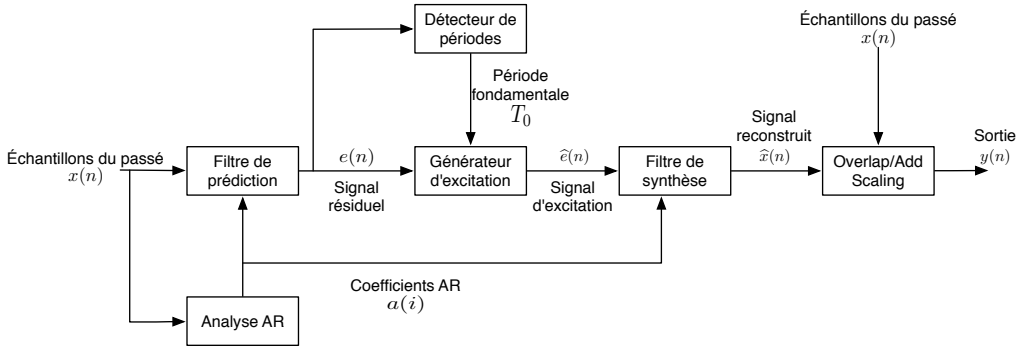


FIGURE 2.1 – Système de masquage de pertes de paquets proposé par Gunduzhan *et al.*

D'autres systèmes de masquage de pertes de paquets ont été proposés sur la base du modèle *harmonique plus bruit*. Dans ce paragraphe, nous présentons les principes de ces deux types de systèmes.

2.2.1 Système de Gunduzhan et al.

Gunduzhan et al. [18] proposent un système de masquage de pertes de paquets reposant sur la prédiction linéaire et un modèle source-filtre. Dans un premier temps, le filtre formantique de coefficients $\{a(i)\}_{i=1,\cdots,P}$ est estimé sur la dernière trame reçue $x(n)$ à l'aide d'un modèle auto-régressif. Le signal résiduel $e(n)$ en est extrait. La période fondamentale T_0 de ce signal résiduel est alors calculée à l'aide d'une méthode par autocorrélation. Cette période sert à générer la trame résiduelle manquante $\widehat{e}(n)$ par une méthode similaire à celle utilisée par l'annexe 1 du G711 (cf. paragraphe 2.1.1). Elle est ensuite filtrée avec le filtre formantique. La trame synthétisée $\widehat{x}(n)$ est alors fondue avec les trames adjacentes par une méthode de superposition / admission. Le schéma global du système proposé par Gunduzhan est représenté sur la figure 2.1.

2.2.2 Utilisation des modèles harmoniques plus bruit

Plusieurs systèmes [19, 20] utilisent un modèle de signal de parole de type "harmoniques plus bruit" :

$$x(t) = \sum_{k=1}^{H} A_k \sin(2\pi k f_0 t + \Phi_k) + b(t) \tag{2.2}$$

dans lequel A_k est l'amplitude de la k^{e} harmonique, H est le nombre d'harmoniques au temps t, f_0 est la fréquence fondamentale, $b(t)$ correspond au bruit du modèle. Les amplitudes, fréquences et phases du signal sont estimées sur la dernière trame reçue. Ces paramètres servent ensuite à la génération de la trame manquante par extension du modèle (2.2). Le bruit additif $b(t)$ est obtenu en filtrant un bruit blanc à l'aide d'un filtre auto-régressif estimé sur la dernière trame connue.

Lindblom et al. proposent dans [21, 22, 23] une extension de l'approche de Gunduzhan présentée dans le paragraphe précédent en modélisant le signal résiduel $e(n)$ à l'aide d'un modèle harmonique plus bruit. Ainsi, $e(n)$ est représenté par une somme de sinusoïdes à laquelle on ajoute une composante aléatoire (du bruit) et sert d'excitation au filtre de synthèse (voir figure 2.1).

2.3 Masquage de pertes de paquets à l'aide de modèles de Markov cachés

L'utilisation de modèles de Markov cachés en reconnaissance de la parole est largement répandue [24]. A ce titre, les modèles de Markov cachés peuvent être considérés comme une modélisation du signal de parole. Toutefois, ils n'ont été utilisés à notre connaissance que dans une seule étude pour le masquage de pertes de paquets. C'est cette étude, publiée par Christoffer Rødbro [25, 26] qui est à l'origine de notre travail et la motivation de nos recherches dans ce domaine.

Ainsi, dans ce paragraphe, après avoir rappelé et précisé les principes et notations des modèles de Markov cachés, nous présentons en détail la

contribution de C. Rødbro dans ce domaine.

2.3.1 Rappels et notations

On considère un système pouvant être décrit à n'importe quel instant t comme étant dans un état particulier parmi Q états. On note q_t la variable aléatoire qui représente le numéro de cet état à l'instant t. La propriété des chaînes de Markov d'ordre 1 dit que la probabilité d'être dans l'état j à l'instant t connaissant tout le passé $\mathrm{P}\left(q_t|q_1 \cdots q_{t-1}\right)$ est égale à celle ne connaissant que l'état à l'instant précédent. Ainsi :

$$\begin{aligned}\mathrm{P}\left(q_t = j|q_1 \cdots q_{t-2}, q_{t-1} = i\right) =& \mathrm{P}\left(q_t = j|q_{t-1} = i\right)\\ =& a_{i,j}\end{aligned}$$

La matrice $\boldsymbol{A} = (a_{ij})_{1\leq i\leq Q, 1\leq j\leq Q}$ est appelée matrice de transition. On note $\boldsymbol{A}^\dagger$ sa matrice transposée.

Le modèle est dit caché lorsque l'état n'est pas observé directement mais au travers d'un processus d'observation. Afin de modéliser ce processus, à chaque état i est associée la loi b_i des observations. Les notations ainsi introduites sont similaires à celles utilisées par Lawrence Rabiner et al. dans le tutoriel [27] sur les modèles de Markov cachés et quelques applications de ces modèles à la reconnaissance de parole.

2.3.2 Principe général

Christoffer Rødbro a développé un système de masquage de pertes de paquets à l'aide de modèle de Markov caché [25]. L'innovation vient à la fois de l'utilisation d'un modèle de Markov pour faire la prédiction ou l'estimation et du fait que c'est l'un des rares systèmes à proposer une approche de minimisation d'erreur.

Nous rappelons que dans ce qui suit, nous supposons que la trame de 1 à $\tau - 1$ ainsi que celles de $\tau + L$ à $\tau + L + J - 1$ sont reçues. Les trames $\tau, \cdots, \tau + L - 1$ sont perdues. De chaque trame de signal est extrait un

vecteur d'observation qui sera noté $\boldsymbol{\phi}_t$ pour la trame t. La perte d'un paquet se traduit alors par l'absence d'une observation, comme l'illustre la figure 2.2

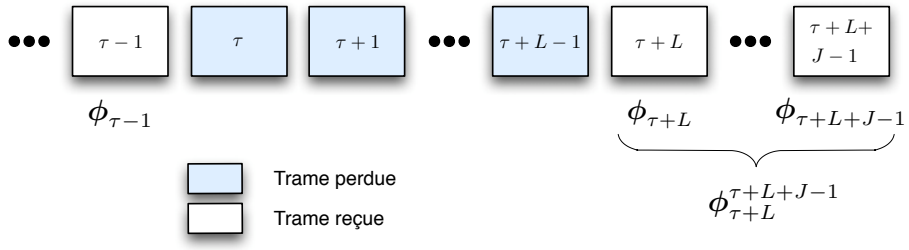


FIGURE 2.2 – Indices des observations et des pertes de paquets.

Christoffer Rødbro propose de modéliser la série temporelle des $\{\boldsymbol{\phi}_t\}_{t=1}^{\tau+L+J-1}$ à l'aide d'un modèle de Markov caché afin de traduire l'évolution de la statistique des observations. Contrairement aux autres méthodes de masquage de pertes basées sur une répétition plus ou moins évoluée de la trame précédente, cette approche permet la prise en compte d'un contexte sonore et linguistique plus important.

Les paragraphes suivants reprennent les étapes principales de l'étude de Rødbro. Les développements des calculs sont présentés dans l'annexe A.

Dans ce qui suit, la série temporelle des observations $\boldsymbol{\phi}_t$ de $t = t_1$ à $t = t_2$ sera notée $\boldsymbol{\phi}_{t_1}^{t_2}$.

Le système de masquage de pertes de paquets proposé par Rødbro repose sur la détermination de la densité de probabilité de l'observation manquante $\boldsymbol{\phi}_{\tau+k}$ connaissant toutes les observations reçues :

$$\mathrm{P}\left(\hat{\boldsymbol{\phi}}_{\tau+k}|\boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \tag{2.3}$$

Or en développant avec la loi de Bayes et en utilisant la propriété de

Markov, (2.3) devient :

$$\mathrm{P}\left(\hat{\boldsymbol{\phi}}_{\tau+k}|\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right)=\sum_{i=1}^{Q}\mathrm{P}\left(\hat{\boldsymbol{\phi}}_{\tau+k},q_{\tau+k}=i|\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \quad (2.4)$$

$$=\sum_{i=1}^{Q}\mathrm{P}\left(\hat{\boldsymbol{\phi}}_{\tau+k}|q_{\tau+k}=i\right)\cdot$$

$$\mathrm{P}\left(q_{\tau+k}=i|\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \quad (2.5)$$

Il devient alors nécessaire de déterminer la probabilité d'être dans l'état i du modèle à l'instant $\tau+k$ connaissant toutes les observations reçues :

$$\mathrm{P}\left(q_{\tau+k}=i|\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right)\qquad 0\leq k<L \quad (2.6)$$

Il est possible d'exprimer (2.6) (les développements sont dans l'annexe A) en fonction des variables forward $\boldsymbol{\alpha}_{\tau-1}$, backward $\boldsymbol{\beta}_{\tau+L}$ et de la matrice de transition $\boldsymbol{A}$ du modèle de Markov :

$k=0,\cdots,L-1$

$$\mathrm{P}\left(q_{\tau+k}=i|\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+LJ-1}\right)=\frac{\left(\boldsymbol{A}^{(k+1)}\boldsymbol{\alpha}_{\tau-1}\right)_i\left(\left(\boldsymbol{A}^{\dagger}\right)^{(L-k-1)}\boldsymbol{\beta}_{\tau+L-1}\right)_i}{\mathrm{P}(\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1})} \quad (2.7)$$

D'après (2.4), la densité de probabilité d'un vecteur manquant est un mélange des lois d'observations du modèle de Markov. Cela peut s'écrire de la manière suivante :

$$\boldsymbol{\phi}_{\tau+k}\sim\sum_{i=1}^{Q}w_ib_i \quad (2.8)$$

où b_i est la loi des observations de l'état i, w_i est le poids du i^{e} état dans l'estimation. Christoffer Rødbro propose d'estimer le vecteur manquant $\hat{\boldsymbol{\phi}}_{\tau+k}$ en minimisant l'espérence de l'erreur quadratique sous cette hypothèse de mélange de lois.

On minimise l'erreur quadratique entre le vecteur à estimer $\hat{\boldsymbol{\phi}}_{\tau+k}$ et le mélange de gaussiennes.

$$MSE(\hat{\boldsymbol{\phi}}_{\tau+k})=E\left[\left\|\boldsymbol{\phi}_{\tau+k}-\hat{\boldsymbol{\phi}}_{\tau+k}\right\|^2\right] \quad (2.9)$$

Dans son étude, les lois d'observations sont des lois normales multivariées. Ainsi b_n est une loi normale de vecteur moyenne $\boldsymbol{\mu_n}$ et de matrice de covariance $\boldsymbol{\Sigma}_n$.

Dans ce cas, le minimum de (2.9) est obtenu pour :

$$\hat{\boldsymbol{\phi}}_{t+k} = \sum_{i=1}^{Q} \mathrm{P}\left(q_{\tau+k} = i | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \boldsymbol{\mu}_i \tag{2.10}$$

Il en ressort que :

$$\hat{\boldsymbol{\phi}}_{t+k} = \sum_{i=1}^{Q} w_i \boldsymbol{\mu}_i \tag{2.11}$$

$$w_i = \mathrm{P}\left(q_{\tau+k} = i | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \tag{2.12}$$

.

2.3.3 Utilisation de la dérivée des observations

Christoffer Rødbro propose [25] également plusieurs améliorations au principe exposé précédemment. La première est de considérer non pas $\boldsymbol{\phi}_t$ comme observation mais $\boldsymbol{\Delta}_t = \boldsymbol{\phi}_t - \boldsymbol{\phi}_{t-1}$. Le problème est alors identique au problème précédent excepté le fait qu'il est nécessaire de raccorder les extrémités du "trou" :

$$\boldsymbol{\phi}_{t+L} - \boldsymbol{\phi}_{\tau-1} = \sum_{k=0}^{L} \boldsymbol{\Delta}_{\tau+k} \tag{2.13}$$

L'ajout d'une telle contrainte complique les calculs, notement pour les cas où plus d'un paquet consécutif est perdu. Christoffer Rødbro montre que cette extension apporte peu de gain en terme de performances.

2.3.4 Lois auxiliaires

Toujours dans son article [25], Christoffer Rødbro pousse un petit peu plus loin sa réflexion en proposant d'utiliser le modèle de Markov comme un moyen de sélectionner la méthode de masquage de pertes de paquets appropriée au son perdu. Ainsi, à chaque état du modèle de Markov, Rødbro associe une

densité de probabilité dite "auxiliaire" qui est adaptée à la génération des trames. Cette densité ne représente plus les observations $\boldsymbol{\phi}$ mais un vecteur lié aux observations qui sera noté $\boldsymbol{\psi}$. Ainsi la stratégie de masquage de pertes est propre à chaque état. Il propose d'utiliser ces densités auxiliaires sous la forme suivante :

$$\mathrm{P}\left(\boldsymbol{\psi}_{\tau+k}|q_{\tau+k}, \boldsymbol{\psi}_{\tau+k-1} \cdots \boldsymbol{\psi}_{\tau+k-K}\right) \tag{2.14}$$

Ainsi la probabilité du vecteur généré (et adapté à la trame en cours) $\boldsymbol{\phi}_{\tau+k}$ à l'instant $\tau + k$ ne dépend plus seulement de l'état du modèle de Markov à cet instant mais également de K précédentes observations.

2.3.5 Implémentations

Christoffer Rødbro utilise un vecteur de paramétrisation basé sur une modélisation "harmoniques plus bruit". Ce vecteur est composé de :

- la fréquence fondamentale,
- de la fréquence de coupure haute des harmoniques,
- du gain du modèle auto-régressif,
- de 10 coefficients Line Spectrum Frequencies (LSF) [28] du-dit modèle.

Ces paramètres sont choisis en fonction du codeur de parole voisé : un codeur sinosoïdal.

De ce fait, le vecteur d'observation est différentié selon que la trame analysée est voisée (présence de fréquence fondamentale et de fréquence de coupure harmonique) ou non.

Deux modèles de Markov sont donc introduits : le premier pour les sons voisés et le second pour les sons non-voisés. Le choix de l'un ou l'autre des modèles semble être lié à la nature de la dernière trame reçue. Cela entraîne l'ajout de probabilités pour passer d'un modèle à l'autre. Le modèle voisé comporte 288 états et le modèle non-voisé en comporte 42 ce qui fait un total de 330 états. Pour le cas où la dérivée (2.13) est utilisée en tant que vecteur d'observation, il est nécessaire de considérer quatre modèles différents :

- voisé/voisé,
- voisé/non-voisé,
- non-voisé/non-voisé,
- non-voisé/voisé.

Les modèles sont appris sur le corpus TIMIT à l'aide d'une première phase d'initialisation (K-Means), puis une phase de raffinement en utilisant l'algorithme de Baum-Welch. Les paramètres des lois auxiliaires sont appris sur le même corpus que celui servant à l'apprentissage des modèles.

Plusieurs améliorations empiriques sont ajoutées au système à l'issu de la phase d'estimation des paramètres manquant :

- l'augmentation du gain est limitée à $1dB$ d'une trame sur l'autre et à $3dB$ sur toute la perte.
- le spectre est modifié soit si les coefficients "Line Spectrum Frequencies" (LSF) ne sont pas dans l'ordre, soit si les pôles du filtre auto-régressif sont trop proches du cercle unité, ou encore si les LSF sortent de l'intervalle $[0, \pi]$.
- la fréquence fondamentale est lissée.

Ces améliorations ont pour but de garder une vecteur à l'entrée de la synthèse qui soit cohérent avec ce qu'il représente : coefficients de prédiction linéaire conduisant à un filtre stable, ni sauts de fréquence fondamentale, ni sauts d'énergie.

2.4 Propositions

Nous proposons dans la suite de ce manuscrit plusieurs améliorations au système publié par Christoffer Rødbro.

Nous souhaitons dans un premier temps nous affranchir de la distinction entre les trames voisées et non-voisées. Nous proposons de réaliser cela à

travers un nouveau vecteur de paramètres incluant un indicateur continu de voisement innovant. Ce nouvel indicateur sera l'objet du chapitre 4.

Une seconde partie de notre étude est consacrée aux méthodes de construction des modèles de Markov servant de support à ces sytèmes de masquage de pertes. A cette occasion, une nouvelle méthode d'estimation des observations manquantes reposant sur le meilleur chemin en l'absence de certaines observations sera proposé. Cette partie est décrite dans le chapitre 5.

La dernière partie est consacrée à l'étude, l'implémentation et la validation dans un système de masquage de pertes de paquets des contributions présentées dans ce manuscrit. Ces résultats sont présentés au chapitre 6.

Toutefois, ces trois chapitres clefs de notre étude nécessitent une base de données sur laquelle seront menées les différentes validations. Ces données sont décrites dans le chapitre suivant.

Chapitre 3

Corpora

Les résultats et différentes évaluations menées tout au long de ce travail utilisent comme support des corpora de parole reconnus. La qualité audio de ces bases de données varie, de la parole enregistrée en environnement de studio, très guidée, à de la parole enregistrée lors de conversations téléphoniques. Dans le cadre de cette étude, trois corpora de parole enregistrés et annotés ont été utilisés :

BREF80 est un corpus de parole lue en français, équilibré phonétiquement et enregistré en studio en 1995 à une fréquence d'échantillonnage de 16kHz.

BREF80BE est identique à BREF80 mais en bande téléphonique (300-3400 Hz).

OGI-MLTS est un corpus de parole téléphonique spontanée comportant plusieurs langues.

Ces trois corpora sont détaillés ci-dessous.

3.1 BREF80

BREF80 [29] est un corpus de lectures d'articles du journal quotidien « *Le Monde* ». Ce corpus a été développé par le Laboratoire d'Informatique pour

la Mécanique et les Sciences de l'Ingénieur (LIMSI) en 1995. Le signal est échantillonné à $16kHz$ en PCM 16 bits. Le niveau de bruit est très faible.

Les articles du journal ont été sélectionnés pour maximiser le nombre de contextes phonétiques. Le corpus représente un vocabulaire de plus de 20000 mots répartis dans 5330 phrases produites par 80 locuteurs. Le tableau 3.1 présente les différents phonèmes présents et étiquetés dans le corpus.

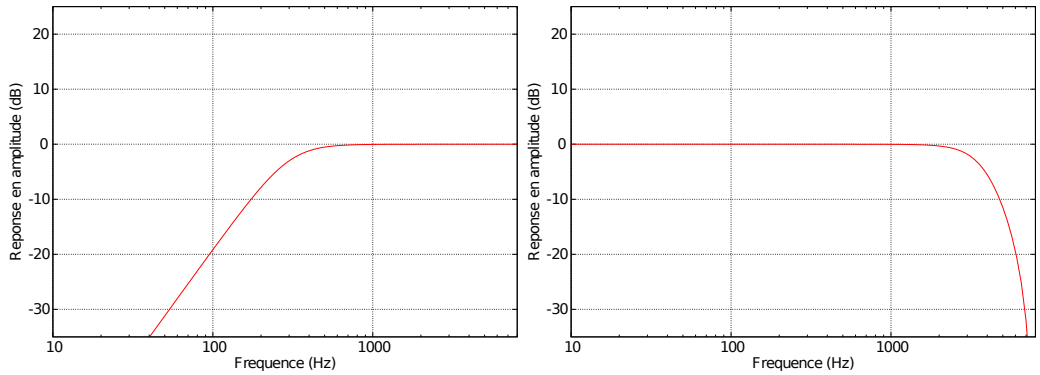
Les annotations phonétiques ont été obtenues à l'aide d'un alignement automatique de la transcription orthographique avec le signal acoustique [30]. Il s'agit, à partir de la transcription orthographique en mots (non alignés temporellement), de générer automatiquement une transcription phonétique à l'aide d'un logiciel de phonétisation. Plusieurs hypothèses de prononciation sont faites pour chaque phrase. La transcription phonétique est alors sélectionnée et alignée sur le signal de parole à l'aide de modèles acoustiques (dans ce cas, des modèles de Markov cachés). Cette méthode permet de disposer d'annotations phonétiques fines pour un grand volume de données ce qui est difficilement réalisable à l'aide d'annotateurs experts.

La répartition de ce corpus en terme de deux sous-ensembles, le premier destiné à l'apprentissage, et le second destiné aux tests, est résumée dans le tableau 3.2. Cette répartition, proposée par le fournisseur de corpus, est commune à toutes les études menées ci-après.

3.2 BREF80BE bande étroite

Nous avons construit le corpus BREF80BE (en bande téléphonique) en dégradant le corpus BREF80 d'origine pour rendre compte des conditions de communications téléphoniques. Ceci est fait, pour des raisons pratiques, à l'aide de deux filtres : un passe-haut à 300Hz et un passe-bas à 3400Hz ; les réponses en fréquences de ces deux filtres IIR d'ordre (2, 2) sont représentées sur la figure 3.1.

Afin de simuler le codeur de parole, une compression de type G.711 [2] (i.e. en loi *mu* du signal) est alors appliquée.



(a) Filtre passe-haut *(b) Filtre passe-bas*

FIGURE 3.1 – Filtrage de passage en bande téléphonique.

Les annotations sont identiques à celles du corpus BREF80.

Le corpus de parole ainsi transformé est alors l'équivalent de BREF80 enregistré à travers un canal téléphonique de bonne qualité. En allant au-delà de la réduction de la bande passante, cela nous permet de mieux rendre compte du type de signaux visé par cette étude : la téléphonie.

3.3 OGI MultiLingual Telephonic Speech

OGI MultiLingual Telephonic Speech [31] est un corpus de parole téléphonique comportant 11 langues : l'anglais, le farsi, le français, l'allemand, l'hindi, le japonais, le koréen, le mandarin, l'espagnol, le tamil et le viétnamien. Il a été développé par le centre de compréhension du langage parlé (CSLU) de l'université de l'Oregon (OGI). Le protocole d'enregistrement[1] a été le suivant : chaque locuteur a appelé un numéro vert aux États-Unis et a dû répondre à une série de questions. Pour six langues (anglais, allemand, hindi, japonais, mandarin et espagnol) il leur a en plus été demandé de raconter une histoire (« *story* ») pendant une minute. Ces histoires ont subi une annotation phonétique très fine (phonèmes et diacritiques) réalisée par des phonéticiens experts selon un même protocole [32].

[1] décrit à l'adresse suivante : `http://www.cslu.ogi.edu/corpora/mlts/protocol.html`

La répartition des durées de signal étiqueté phonétiquement par langue est présentée dans le tableau 3.3.

Ce corpus n'a pas été divisé en deux sous-ensembles car le volume de chaque sous-ensemble, notamment celui du sous-ensemble d'apprentissage, aurait été trop faible pour avoir une signification statistique et permettre l'estimation de paramètres. Néanmoins, l'intégralité de ce corpus est mise à contribution dans le chapitre 4.

3.4 Utilisation des différents corpora

Les différents corpora présentés dans ce chapitre servent aux différentes évaluations des méthodes et outils mis en œuvre dans ce document. Les sous-ensembles d'apprentissage sont utilisés pour réaliser l'estimation des paramètres internes des systèmes ; leurs performances sont ensuite évaluées sur le sous-ensemble de test.

Ainsi, BREF80 et OGI_MLTS servent de support à l'évaluation d'un nouveau degré de voisement, le pourcentage de voisement, qui sera présenté dans le chapitre suivant.

Le sous-ensemble anglais de OGI_MLTS et l'ensemble d'apprentissage de BREF80BE sont utilisés pour l'estimation des systèmes de masquage de paquets. La partie "Test" de BREF80BE nous permet d'évaluer et de comparer les performances des systèmes développés.

Symbole	Phonétique	Exemple	Classe
A	/a/	pl**a**t	Voyelle
I	/i/	p**i**le	
U	/y/	r**u**e	
O	/ɔ/	enc**o**re, f**o**rte, al**o**rs	
AU	/o/	b**eau**, t**ô**t, s**eau**	
EU	/ø/	p**eu**, l**e**	
OE	/œ/	h**eu**re	
EI	/e/	bl**é**	
AI	/ɛ/	l**ai**t	
OU	/u/	r**ou**e	
AN	/ã/	bl**an**c	Voyelle nasale
ON	/ɔ̃/	b**on**	
IN	/œ̃/	**in**, br**un**, b**ain**	
Y	/j/	act**ion**, b**ie**n, h**ie**r	Semi-consonne
W	/w/	b**oi**te, **we**b, **oui**, c**oin**	
L	/l/	**l**ent	Liquide
R	/ʁ/	**r**ue	
M	/m/	**m**ot	Nasale
N	/n/	**n**ous	
NG	/ŋ/	camp**ing**	
F	/f/	**f**er	Fricative sourde
S	/s/	a**ss**is	
CH	/ʃ/	**ch**ou	
V	/v/	**v**erre	Fricative sonore
Z	/z/	As**i**e	
J	/ʒ/	**j**oue	
P	/p/	**p**eu	Occlusive sourde
T	/t/	**t**on	
K	/k/	**c**ou	
B	/b/	**b**asse	Occlusive sonore
D	/d/	**d**oux	
G	/g/	**g**oût	
SIL		Silence	Silence
SP		Court silence entre deux mots	

TABLE 3.1 – Phonèmes étiquetés dans le corpus BREF80.

Partie	**Sexe**	**Durée**
Apprentissage	homme	4 :33 :12
	femme	5 :41 :45
	total	10 :14 :57
Test	homme	0 :27 :46
	femme	0 :29 :58
	total	0 :57 :44

TABLE 3.2 – Répartition temporelle du corpus BREF80.

Langue	**Durée totale**
anglais	2 h 2 mn
allemand	1 h 24 mn
espagnol	1 h 30 mn
hindi	56 mn
japonais	53 mn
mandarin	58 mn

TABLE 3.3 – Durée totale des fichiers disposant d'une transcription phonétique sur OGI-MLTS.

Chapitre 4

Pourcentage de voisement

Le signal de parole peut être divisé en trois grands types de zones traduisant trois types de production de la parole :

- Les zones de non-parole qui sont de très faible énergie (fermeture du conduit vocal, fin de phrase).

- Les sons voisés caractérisés par la présence d'une fréquence fondamentale induite par la vibration des cordes vocales. Ces zones sont généralement de forte énergie comme par exemple dans le son /ɔ/ (dans r**o**seau /ʁɔzɔ/). La figure 4.1-(a) présente le spectre du /ɔ/ de r**o**seau. On peut y remarquer la présence de la fréquence fondamentale ainsi que celle de ses harmoniques.

- Les sons non-voisés où toute périodicité est absente comme dans le son /ʃ/ (dans **ch**assis, /ʃasi/). On remarque l'absence de fréquence fondamentale et d'harmonique ainsi qu'une plus faible énergie sur la figure 4.1-(b).

Le voisement se traduit après analyse selon différents attributs :

- Un indice binaire pour indiquer la présence ou l'absence du trait de voisement,

- Une valeur de la fréquence fondamentale, mise à 0 dans les zones non voisées (valeur continue sur $\mathbb{R}$ et égale à 0 par morceaux),

- Un degré de voisement, valeur comprise entre 0 et 1 traduisant un degré d'harmonicité : si le degré de voisement est égal à 1, le signal est périodique ; inversement, si il est nul, le signal est apériodique.

L'utilisation d'indices binaires comme information de voisement pose des problèmes du fait que ces attributs ne sont pas continus ou qu'ils sont employés juxtaposés à des attributs de nature continue ; c'est le cas lorsqu'ils deviennent observations dans les modèles de Markov cachés.

Dans ce type d'application, il est alors plus simple d'utiliser deux modèles, un pour la partie voisée et un autre pour la partie non voisée. Mais il est alors nécessaire de gérer les transitions entre ces modèles à l'aide de règles spécifiques, problème qui peut s'avérer très difficile si l'on pense au seul phénomène d'assimilation du trait de voisement.

Une représentation continue du voisement a comme avantage d'être, tout en étant proche de la linguistique, un paramètre facilement utilisable dans des domaines comme la reconnaissance de parole ou la prédiction, mais elle reste non homogène aux coefficients traditionnels.

Cherchant d'une part à introduire un degré de voisement pour contourner le problème des deux modèles de Markov cachés proposés par C. Rødbro et d'autre part à garder l'efficacité de ce type de modèle, nous définissons et évaluons l'intérêt d'un nouveau paramètre, que nous appelons "pourcentage de voisement".

Dans ce chapitre, nous examinons les principales méthodes existantes incluant la notion de degré de voisement et nous détaillons la méthode proposée pour l'obtention de ce paramètre. Son utilisation est dans un premier temps validée dans une application de segmentation voisée/non-voisée. Puis l'intérêt de ce paramètre est évalué dans le cadre d'un système de reconnaissance phonétique.

4.1 État de l'art du "degré de voisement"

Le degré de voisement apparaît comme paramètre intermédiaire pour détecter les zones voisées afin de calculer la fréquence fondamentale. Il est lié à la recherche de périodicités dans un signal. En effet, en traitement de la parole, le degré de voisement est rarement estimé de manière directe. Il est plutôt un sous-produit d'algorithmes qui ont pour but essentiel d'estimer la fréquence fondamentale, le degré de voisement étant alors un degré de confiance dans la mesure du pitch. Compte tenu de l'importance de la fréquence fondamentale en parole comme en musique, de nombreux algorithmes ont vu le jour pour répondre à ce problème spécifique [33] ; ils s'appuient pour nombre d'entre eux sur des calculs de corrélation, que ce soit dans le domaine temporel ou dans le domaine fréquentiel. Nous donnons ci-après trois exemples fondamentaux.

4.1.1 Fonctions aux différences

L'algorithme du YIN [14] est un exemple d'un algorithme de détection de la fréquence fondamentale d'un signal audio, qu'il s'agisse de parole ou de musique. L'estimation de la période fondamentale repose sur la fonction de différences cumulées $d'(\theta)$ définie de la manière suivante :

$$d'(\theta) = \begin{cases} 1 & \text{pour } \theta = 0 \\ d(\theta) / \left[(1/\theta) \sum_{j=1}^{\theta} d(j) \right] & \text{pour } \theta > 0 \end{cases} \tag{4.1}$$

$$d(\theta) = \sum_{t=-\infty}^{\infty} \left[(x(t) - x(t-\theta) \right]^2 \tag{4.2}$$

La valeur de la période fondamentale est l'indice θ correspondant à un minimum local de la fonction $d'(\theta)$. La valeur de ce minimum, lorsque la fréquence fondamentale est définie, peut être interprétée comme un degré de voisement. Il représente la confiance de l'estimateur sur la fréquence estimée : plus elle est basse, plus la "trame est voisée" et la valeur de la période est correcte.

La trame analysée est considérée comme voisée si la valeur de $d'(\theta)$ passe en dessous de 0.1 sur la fenêtre de calcul de d' [14].

4.1.2 Corrélation temporelle

L'idée est la suivante : si le signal est périodique de période θ_0, alors la valeur de la fonction d'autocorrélation R en θ_0, $R(\theta_0)$, est la puissance associée à la fréquence fondamentale.

Le degré de voisement ou coefficient de périodicité est alors défini comme le maximum de la fonction d'autocorrélation normalisé par la puissance du signal. Ce maximum est recherché dans une plage temporelle autour de la période fondamentale.

Le coefficient de périodicité C_p de la trame analysée s'écrit de la manière suivante :

$$C_p = \frac{\max\limits_{T_{min}<i<T_{max}} R(i)}{R(0)} \tag{4.3}$$

où T_{min} est la période correspondant à la plus grande valeur de fréquence fondamentale attendue alors que T_{max} est celle correspondant à la plus petite fréquence fondamentale que l'algorithme est susceptible de détecter. Cette plage temporelle est absolument nécessaire pour éviter de trouver des multiples ou sous-multiples, mais elle reste un paramètre critique de l'algorithme.

4.1.3 Corrélation mixte

Cho et al. proposent dans [34, 35] de déterminer la fréquence fondamentale d'une trame de parole à l'aide de l'autocorrélation spectro-temporelle $\zeta(\theta)$. Celle-ci est définie à l'aide de l'autocorrélation temporelle $\zeta^T(\theta)$ et de l'autocorrélation fréquentielle $\zeta^F(\omega_\theta)$:

$$\zeta^T(\theta) = \frac{\sum_{n=0}^{N-\theta-1} \tilde{x}(n) \cdot \tilde{x}(n+\theta)}{\sqrt{\sum_{n=0}^{N-\theta-1} \tilde{x}^2(n) \cdot \sum_{n=0}^{N-\theta-1} \tilde{x}^2(n+\theta)}} \tag{4.4}$$

$$\zeta^F(\omega_\theta) = \frac{\sum_{\omega=0}^{N-\omega_\theta-1} \tilde{X}(\omega) \cdot \tilde{X}(\omega+\omega_\theta)}{\sqrt{\sum_{\omega=0}^{N-\omega_\theta-1} \tilde{X}^2(\omega) \cdot \sum_{\omega=0}^{N-\omega_\theta-1} \tilde{X}^2(\omega+\omega_\theta)}} \tag{4.5}$$

où $\omega_\theta = 2\pi\frac{N}{\theta}$, $\tilde{x}$ est la version centrée du signal d'entrée x, $\tilde{X}$ est la version centrée du module de la transformée de Fourier X du signal x. L'autocorrélation $\zeta(\theta)$ est ainsi définie par :

$$\zeta(\theta) = \lambda\zeta^T(\theta) + (1-\lambda)\zeta^F(\omega_\theta) \tag{4.6}$$

avec $0 < \lambda < 1$. La valeur de λ permet de donner une prépondérance soit à la périodicité temporelle, soit la périodicité fréquentielle. Dans la pratique, la valeur de λ est souvent choisie égale à 0.5 afin de ne pas privilégier l'une ou l'autre des approches. En effet, d'après [35], la valeur $\lambda = 0.5$ est celle qui minimise l'erreur d'estimation de la fréquence fondamentale.

La valeur T_0 de la période fondamentale de la trame analysée est donc donnée par :

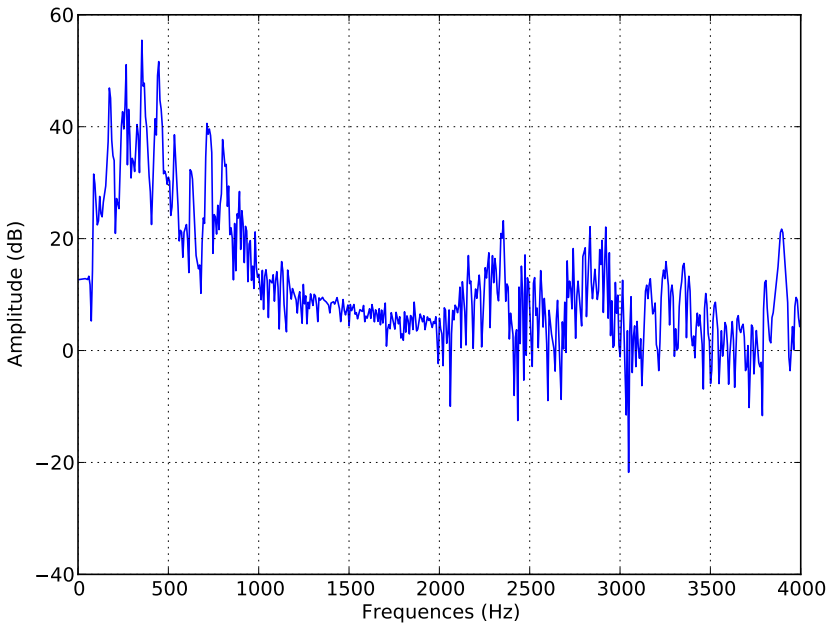
$$T_0 = \underset{\theta}{\operatorname{argmax}}\, \zeta(\theta) \tag{4.7}$$

C'est à partir de cette définition que Cho et al. introduisent [36] le coefficient harmonique H_a et le définissent de la manière suivante :

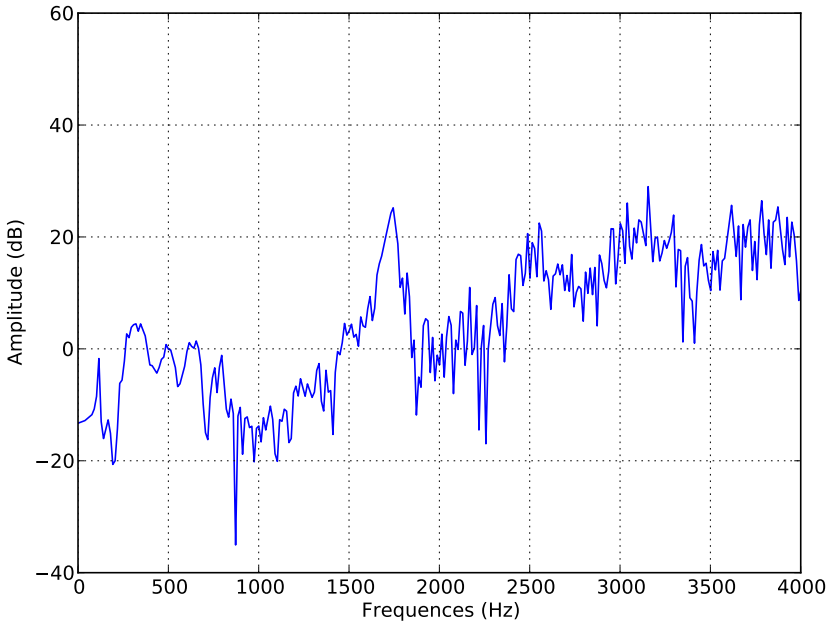
$$H_a = \max_{\theta} \zeta(\theta) \tag{4.8}$$

On remarque que si le coefficient harmonique est faible, peu de confiance est accordée à la valeur trouvée de l'estimateur de fréquence fondamentale, la trame est donc du bruit. Au contraire, si la valeur de H_a est élevée, la

trame analysée est estimée périodique. Ce coefficient harmonique peut ainsi être vu comme un degré de voisement.



*(a) - Voisée (/ɔ/ dans r**o**seau)*



*(b) - Non voisée (/ʃ/ dans **ch**assis)*

FIGURE 4.1 – Densité spectrale de puissance d'une trame de 20ms de signal de parole, (a) voisée, (b) non-voisée.

4.2 Définition du pourcentage de voisement

Dans notre étude, nous souhaitons introduire un paramètre continu mesurant le degré de voisement, sans toutefois le lier nécessairement à l'estimation de la fréquence fondamentale comme cela est fait dans les algorithmes existants présentés dans la section précédente. Nous proposons ainsi, dans cette section, une mesure continue et normée du degré de voisement.

Pour cela, le signal de parole est modélisé comme une somme d'une partie périodique (voisée) $x_V(t)$, et d'une partie non-harmonique $b(t)$:

$$x(t) = x_V(t) + b(t) \tag{4.9}$$

Nous définissons le pourcentage de voisememement $v_{\%}$ comme le rapport entre la puissance de la partie harmonique du signal $P_{\text{harmonique}}$ et la puissance totale P du signal [37]. Ainsi :

$$v_{\%} = \frac{P_{\text{harmonique}}}{P} \tag{4.10}$$

4.2.1 Estimation du pourcentage de voisement

Le pourcentage de voisement défini par (4.10) nécessite l'estimation de deux puissances, la puissance totale du signal et la puissance de sa partie harmonique.

Nous proposons d'estimer ces deux puissances à partir de la densité spectrale de puissance (DSP) du signal $x(t)$, en s'inspirant des travaux de M. Durnerin [38]. L'idée principale consiste à appliquer un filtre médian directement sur le spectre de puissance de la trame étudiée afin de lisser les pics correspondant aux harmoniques. Ceci permet d'estimer la densité spectrale de puissance de la partie non-harmonique du signal, $b(t)$.

Ainsi, la puissance de la partie harmonique du signal, $P_{\text{harmonique}}$, est obtenue en soustrayant à la puissance totale de chaque trame d'analyse, la puissance de la partie non-harmonique.

Le réglage de la taille de la fenêtre du filtre médian est un paramètre crucial. L'étude de Mathieu Durnerin montre que pour une bonne élimina-

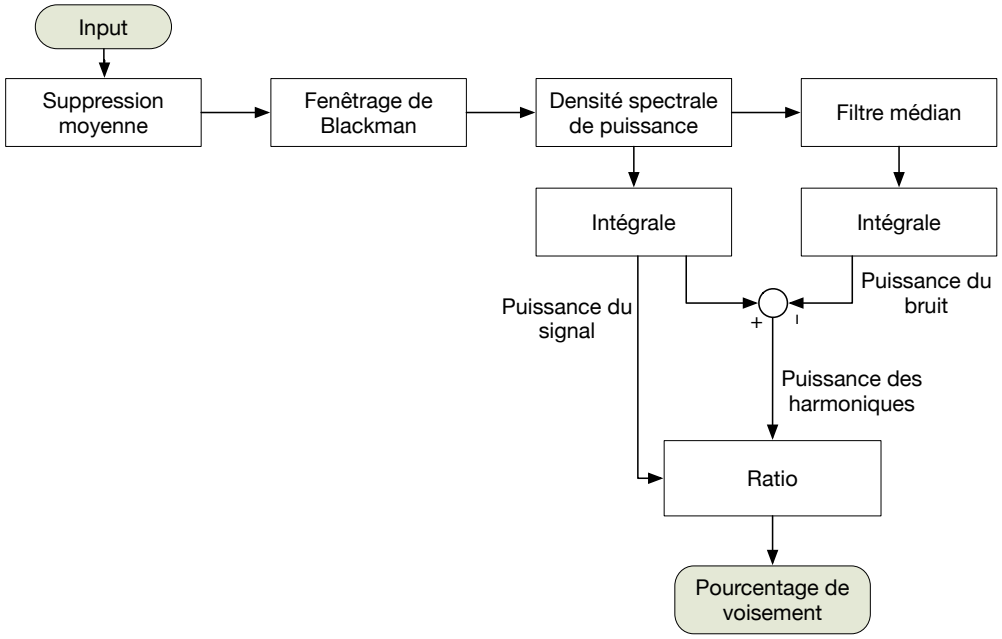


FIGURE 4.2 – Schéma récapitulatif du pourcentage de voisement

tion des pics spectraux correspondant à la fréquence fondamentale et aux harmoniques, la taille de la fenêtre médian doit être supérieure à trois fois la largeur du pic harmonique (largeur induite par la méthode choisie d'estimation spectrale) mais de préférence inférieure à six fois cette largeur afin de rester "local" et de pouvoir suivre les évolutions continues du spectre.

4.2.2 Détails de l'algorithme

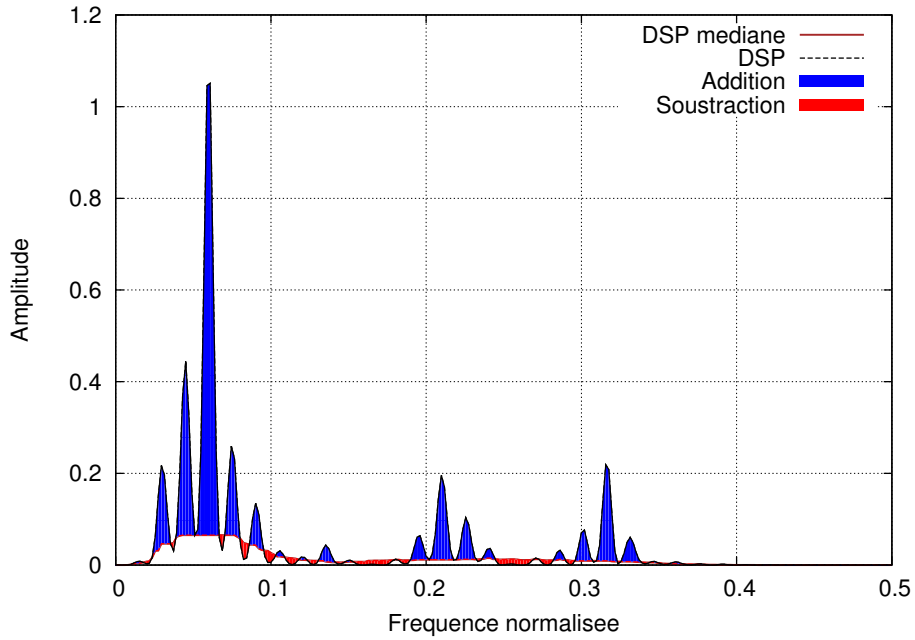
La figure 4.2 fait un résumé de tout le processus mis en œuvre pour le calcul du pourcentage de voisement : la densité spectrale de puissance (DSP) X de chaque trame est calculée sur le signal auquel nous avons retranché la moyenne (noté $\tilde{x}$). Un filtre médian de longueur supérieure à trois fois mais inférieure à six foix la largeur d'un lobe fréquentiel est appliqué directement sur cette densité X. Cela permet d'estimer la ligne de fond qui correspond au spectre de la partie non-harmonique du signal. En intégrant ces deux densités on obtient respectivement la puissance de la partie non-harmonique

et la puissance du signal. La puissance de la partie harmonique est alors définie comme la différence entre ces deux puissances. Le pourcentage de voisement est le rapport entre ces deux quantités et s'exprime ainsi :

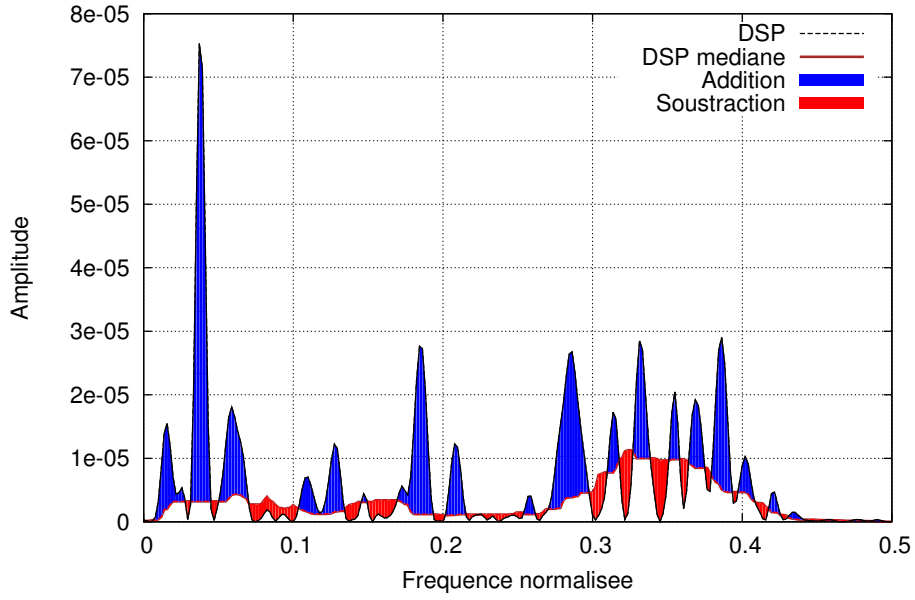
$$v_{\%} = \frac{\int_0^{0.5} \left(X(\tilde{f}) - \text{median}[X](\tilde{f}) \right) \mathrm{d}\tilde{f}}{\int_0^{0.5} X(\tilde{f}) \mathrm{d}\tilde{f}} \tag{4.11}$$

où $\tilde{f} = \frac{f}{f_s}$ est la fréquence normalisée (f_s et f sont respectivement la fréquence d'échantillonnage et la fréquence en Hz), $\tilde{x}$ est le signal analysé auquel nous avons retranché sa moyenne, X est la densité spectrale de puissance de $\tilde{x}$, median est le filtre médian.

La figure 4.3 reprend ces opérations sur une trame voisée (4.3-(a)) et sur une trame non-voisée (4.3-(b)).



(a) Trame voisée ($v_{\%} = 0.69$)



(b) Trame non-voisée ($v_{\%} = 0.51$)

FIGURE 4.3 – Calcul du pourcentage de voisement.

Pourcentage de voisement calculé sur un signal de 240 *echantillons échantillonné à* $f_s = 8kHz$. *La denstité spectrale de puissance est calculée sur* 512 *points et la longueur du filtre médian est de* 25 *points fréquentiels.*

4.3 Protocole d'évaluation

Afin de valider ce nouveau degré de voisement, nous avons étudié la situation classique d'utilisation de ce paramètre, à savoir l'estimation du voisement dans les procédures de segmentation de la parole en zones voisées et en zones non-voisées. Nous avons comparé les performances à celles obtenues par l'algorithme YIN.

Compte tenu de notre objectif, le pourcentage de voisement a également été évalué comme paramètre d'entrée d'un décodeur acoustico-phonétique basé sur un modèle de Markov caché afin de valider sa pertinence et le potentiel risque de dégradation lié à une non-homogénéité des paramètres d'observation.

4.3.1 Segmentation voisée/non-voisée

Voisé, non-voisé, silence

La tâche consiste à segmenter à l'aide d'un système automatique un flux de parole en :

- silence,
- parole voisée,
- parole non-voisée.

Deux approches sont envisagées :

- La décision est prise en deux temps. Tout d'abord la puissance de la trame est comparée à un seuil λ_1 pour déterminer si c'est du silence ou de la parole. Si c'est de la parole, la valeur de l'estimateur continu de voisement est comparée à un deuxième seuil λ_2 afin de déterminer si cette trame est voisée ou non. Ainsi, chaque trame (typiquement une trame d'analyse dure 20ms et est calculée toutes les 10ms) est classée dans l'une des trois catégories citées ci-dessus. Cette approche

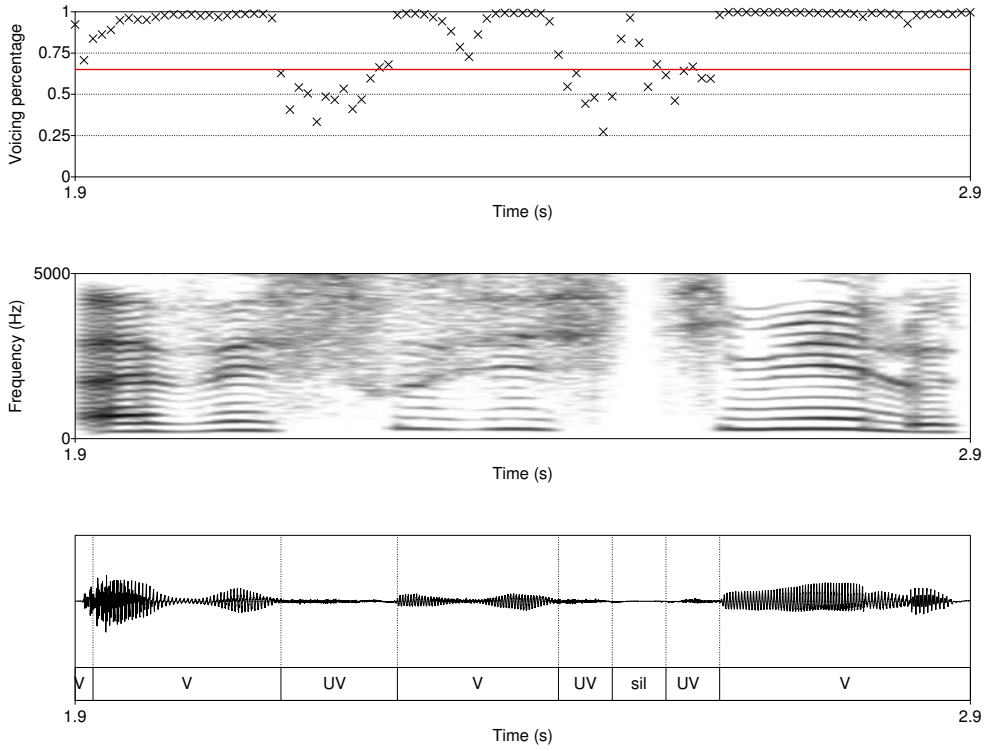


FIGURE 4.4 – Exemple de segmentation voisée / non-voisée / silence. *Dans cet exemple* $\lambda_1 = 3 \cdot 10^{-7}$ *et* $\lambda_2 = 0.65$.

est notée $\mathcal{A}$ dans les tableaux de résultats. Un exemple est présenté sur la figure 4.4. La figure du haut présente la valeur du pourcentage de voisement, celle du milieu est une représentation du spectrogramme, celle du bas montre la segmentation produite par l'introduction du seuil (ligne rouge) sur le pourcentage de voisement ainsi que la forme d'onde.

- Le signal de parole est segmenté *a priori* en zones quasi-stationnaires [39] et ces segments sont classés en se basant sur la valeur du pourcentage de voisement sur la trame supposée la plus stable : les 20ms au milieu du segment quasi-stationnaire. Cette méthode de décision est dite *a posteriori* et est notée $\mathcal{B}$ dans les tableaux de résultats. La figure 4.5 montre de haut en bas, sur le même extrait que la figure 4.4, la segmentation

a priori, la valeur du pourcentage de voisement, le spectrogramme et, les décisions qui ont été prises sur chaque segment.

Mesures de performances

Afin de mesurer et de comparer les performances de segmentation des divers algorithmes d'estimation de voisement, nous avons utilisé des mesures couramment utilisées notamment dans les campagnes du NIST[1]. La mesure utilisée comme référence, celle donnée dans les tableaux de résultats, est le taux d'erreurs de segmentation défini de la manière suivante :

$$R = \frac{\text{durée incorrectement segmentée}}{\text{durée totale à segmenter}} \tag{4.12}$$

[1] `http://www.itl.nist.gov/iad/mig/tools/`

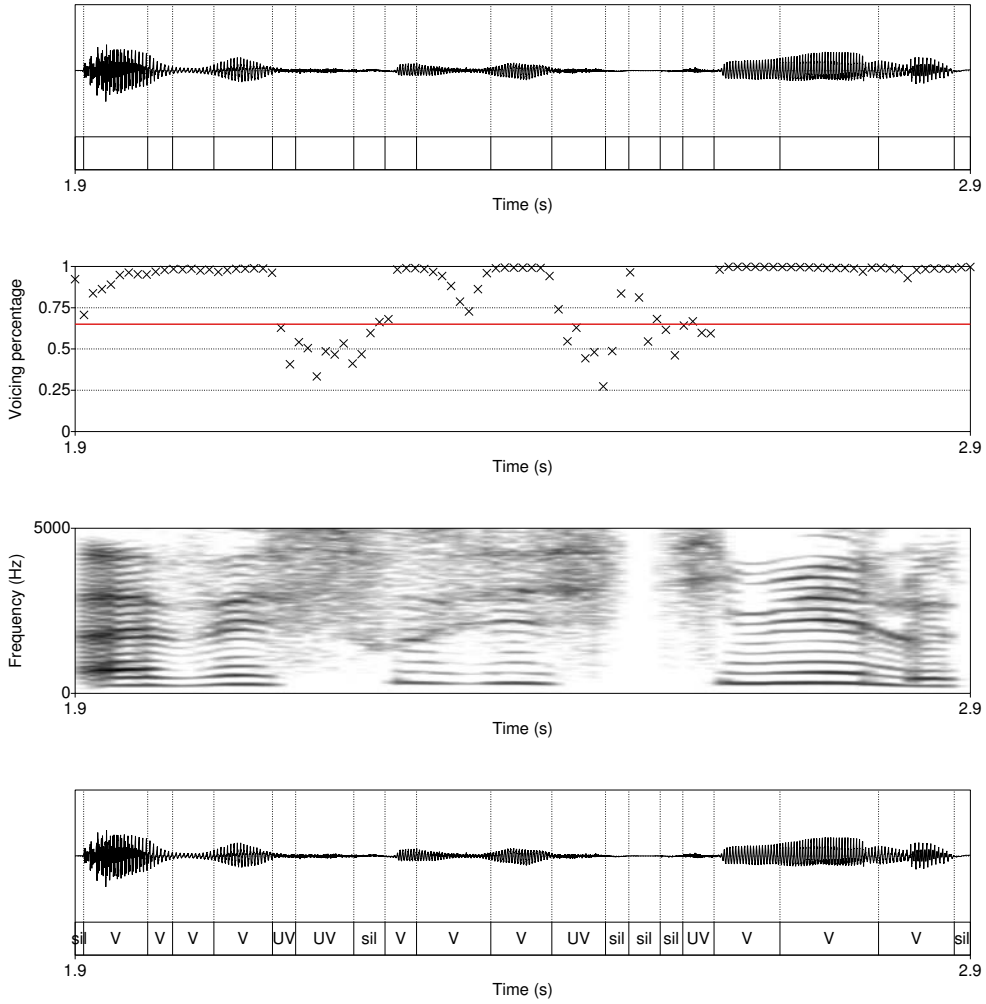


FIGURE 4.5 – Exemple de décision voisée / non-voisée / silence sur une segmentation *a priori*.

Dans cet exemple $\lambda_1 = 3 \cdot 10^{-7}$ *et* $\lambda_2 = 0.65$.

4.3.2 Décodeur acoustico-phonétique

Le pourcentage de voisement est conçu pour être utilisé comme un paramètre supplémentaire d'observation du signal de parole et être intégré dans un modèle de Markov caché, ceci dans le but d'assurer ultérieurement la continuité et le suivi du voisement en phase d'estimation. L'ajout d'un nouveau paramètre non-homogène aux observations ne conduit pas toujours à l'amélioration des performances du modèle dans le cadre de la reconnaissance de parole. C'est pourquoi il est important de valider l'apport du pourcentage de voisement dans ce contexte.

Afin d'évaluer la pertinence de ce nouveau paramètre dans ce cadre, nous avons conçu trois systèmes de reconnaissance automatique des *phones* en faisant varier uniquement la composition des différents vecteurs d'observations et nous avons comparé leurs performances. Nous avons utilisé des paramétrisations présentes dans l'état de l'art du décodage acoustico-phonétique auxquelles nous avons adjoint le pourcentage de voisement. Trois systèmes ont ainsi été développés dans l'optique de comparer leurs performances. Pour chacun de ces systèmes, la structure des modèles de Markov est identique, seule la paramétrisation change. Les trois familles de vecteurs d'observation sont :

MFCC_E_D_Z : 12 Mel Frequencies Cepstral Coefficients [24] (MFCC), l'énergie, les dérivées d'ordre 1. Une soustraction de la moyenne spectrale est effectuée.

LPCC_E_D_Z : 12 Coefficients cepstraux issus de la prédiction linéaire [24] (LPCC), l'énergie, les dérivées d'ordre 1. Une soustraction cepstrale est effectuée.

LPCC_E_D_Z + $V_{\%}$: même paramétrisation que pour LPCC_E_D_Z mais avec en plus le pourcentage de voisement.

Notons que nous aurions pu ajouter une quatrième famille de vecteurs composée des mêmes paramètres que MFCC_E_D_Z à laquelle on ajoute le

pourcentage de voisement. Comme les performances de LPCC_E_D_Z sont supérieures à celles de MFCC_E_D_Z, nous avons opté pour ne réaliser l'expérience qu'avec la famille la plus performante.

Le système de décodage acoustico-phonétique a été développé sur le corpus BREF80. Les 35 phones du français [40] détaillés dans le tableau 3.1 page 31 ont été modélisés indépendamment du contexte. Chacun de ces éléments acoustiques est représenté par un modèle de Markov à trois états.

4.4 Résultats

4.4.1 Segmentation voisée/non-voisée

Influence du seuil de décision λ_2 La performance de la segmentation voisée/non-voisée dépend de la valeur du seuil λ_2. Afin d'en choisir la valeur, nous avons comparé la segmentation produite pour différentes valeurs de seuil avec une segmentation basée sur la détection de la fréquence fondamentale. Cette étude a été réalisée sur le corpus OGI_MLTS présenté dans le chapitre 3, paragraphe 3.3. Les performances du pourcentage de voisement dans le cas $\mathcal{A}$ en fonction du seuil et de la langue sont présentées sur la figure 4.6. Nous constatons que la valeur optimale du seuil est entre $\lambda_2 = 0.55$ et 0.6, selon la langue.

Afin de mettre en évidence la répartition des fausses alarmes, nous avons tracé sur la figure 4.7 l'histogramme des valeurs du pourcentage de voisement lors de fausses alarmes.

On s'aperçoit que quelle que soit la valeur du seuil, celui-ci sépare les deux types d'erreur :

$$P_{\mathrm{V}\to\mathrm{UV}} = \mathrm{P}\,(\text{décider voisé}|\text{le son est non-voisé}) \tag{4.13}$$

$$P_{\mathrm{UV}\to\mathrm{V}} = \mathrm{P}\,(\text{décider non-voisé}|\text{le son est voisé}) \tag{4.14}$$

Si le paramètre proposé $v_{\%}$ n'avait pas été représentatif du degré de voisement d'une trame, la distribution des valeurs du pourcentage de voisement lors d'erreurs n'aurait pas été maximum sur la valeur du seuil fixé.

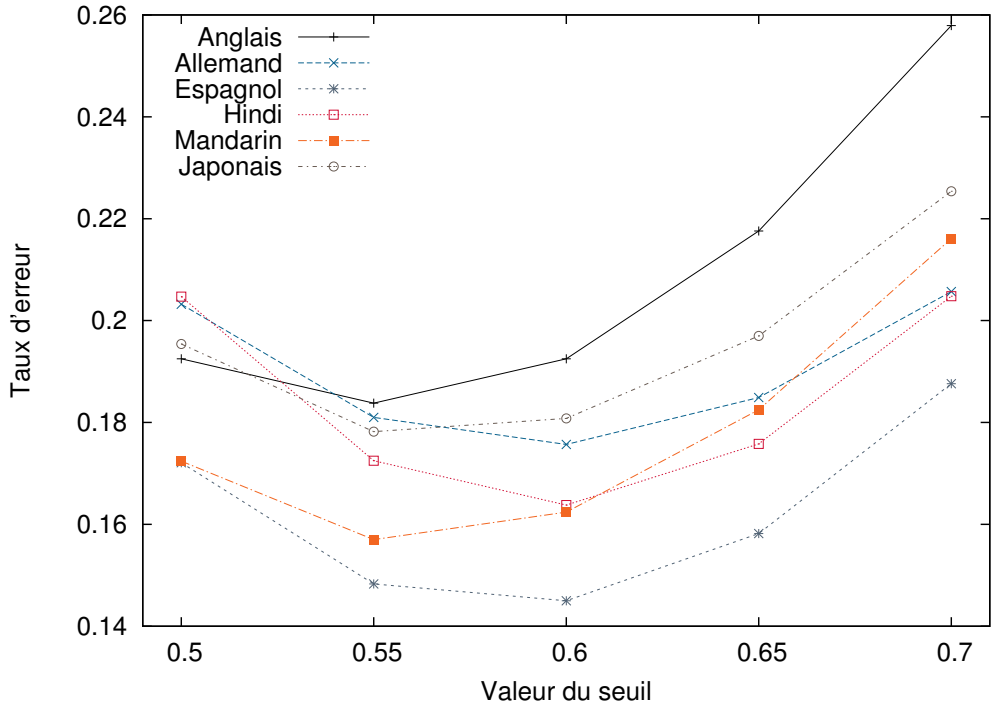


FIGURE 4.6 – Influence du seuil sur la décision de voisement.
Expérimentation effectuée sur les siw langues présentes dans le corpus `OGI_MLTS`.

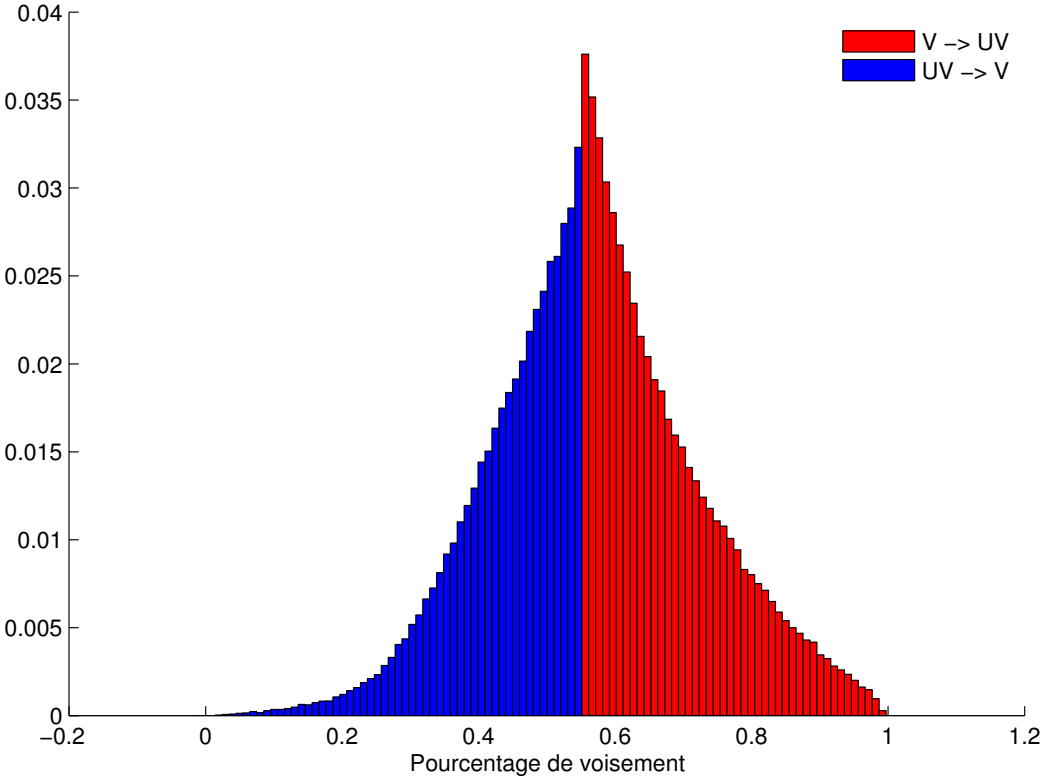


FIGURE 4.7 – Histogramme de fausse segmentation V/UV.
Valeur de seuil : 0.55

Corpus	Langage	$\mathcal{A}$	$\mathcal{B}$
OGI	Anglais	19.34	16.57
	Allemand	17.83	15.47
	Espagnol	16.46	14.45
	Hindi	15.59	14.35
MLTS	Mandarin	16.19	14.22
	Japonais	16.14	15.04
	Tout	17.00	15.10
BREF80	Français	10.74	13.77

TABLE 4.1 – Taux d'erreur de segmentation.

Rapport en % entre le temps incorrectement segmenté (en accord avec la segmentation produite par le YIN) et le temps total.

Cette répartition attendue des deux types d'erreur confirme l'intérêt et la pertinence du paramètre proposé $v_{\%}$ en tant que degré de voisement.

Évaluation des segmentations produites par $\mathcal{A}$ et $\mathcal{B}$ Le tableau 4.1 présente l'erreur de segmentation (4.12) pour le système $\mathcal{A}$ et pour l'approche $\mathcal{B}$. Ces résultats sont à tempérer. En effet, l'annotation est le fruit d'un algorithme performant mais néanmoins automatique de détection de fréquences fondamentales : le YIN. Le taux d'erreurs présenté est donc lié à la fois aux erreurs d'annotations et à celles d'estimation. Toutefois, nous pouvons noter que l'algorithme du YIN [14], d'après son auteur, n'a qu'une erreur d'estimation de 2% de la fréquence fondamentale ce qui conduit à une détection fiable des zones présentant une fréquence fondamentale, donc des segments de parole voisée.

Nous remarquons, au regard des résultats présentés dans le tableau 4.1, que les performances du pourcentage de voisement sont d'ordre de grandeur similaire sur les deux corpus de tests. Dans le cas d'un signal bruité (cas du corpus OGI_MLTS), les performances sont meilleures s'il est utilisé conjoin-

tement à une pré-segmentation.

4.4.2 Décodeur acoustico-phonétique

Les performances des trois systèmes sont mesurées en terme de taux de reconnaissance et "*accuracy*" tels que définis dans le HTK Book [41] :

$$\%Corr = \frac{H}{N} \tag{4.15}$$

$$Acc = \frac{H - I}{N} \tag{4.16}$$

où $H = N - D - S$ est le nombre d'étiquettes correctes, I est le nombre d'insertions, N le nombre d'étiquettes à trouver, D le nombre d'étiquettes omises et S celles substituées.

Modèle	**Acc**	**%Corr**[2]
MFCC_E_D_Z	58.9 (±0.5)%	66.2 (±0.5)%
LPCC_E_D_Z	59.9 (±0.5)%	67.9 (±0.5)%
LPCC_E_D_Z + $V_{\%}$	60.3 (±0.5)%	68.4 (±0.5)%

TABLE 4.2 – Taux de reconnaissance phonétique.

Le tableau 4.2 nous permet de remarquer que l'adjonction du pourcentage de voisement au vecteur de paramétrisation ne dégrade pas les performances du système mais tend à les améliorer très légèrement. Cette légère amélioration est en soi un très bon résultat si on prend en compte qu'en général l'introduction d'un paramètre de nature différente aux autres coordonnées du vecteur d'observation peut induire des pertes de performances.

4.4.3 Discussions

Nous avons introduit une nouvelle mesure de faible coût calculatoire du degré de voisement d'un signal de parole. Les performances de ce degré de voisement dans une tâche de segmentation voisée/non-voisée sont comparables à

[2]Taux de reconnaissance phonétique

celles d'un estimateur de fréquence fondamentale. Cela nous confirme que la quantité mesurée est bien le degré d'harmonicité du signal. Ces performances dans le cadre d'une reconnaissance phonétique, prémices de la reconnaissance de la parole, sont encourageantes et nous confortent dans l'utilisation de ce paramètre afin de représenter le voisement dans un modèle de Markov caché.

Chapitre 5

Modélisation et estimation

Dans le cadre de ce travail, à l'image de celui effectué par Rødbro, nous avons choisi de modéliser l'évolution temporelle des paramètres acoustiques du signal de parole par un modèle de Markov caché (MMC).

L'approche par modèles de Markov cachés a, depuis plus de 30 ans, fait ses preuves dans le monde du traitement de la parole [24, 42]. Dans ce modèle, chaque état représente une brique élémentaire de l'acoustique, un sous-élément d'un *phone* : chaque état représente un son, un segment quasi-stationnaire du signal de parole et les lois portées par les états rendent compte de la distribution probabiliste des trames de signal appartenant à ce segment. Ces successions d'états modélisent l'évolution temporelle du signal acoustique, et ce à différents niveaux : de manière simpliste, les lois des états d'un même phonème sont liées au phénomène de coarticulation avec les phonèmes adjacents, les liaisons possibles entre phonèmes révèlent la structure phonotactique de la langue tandis que les liaisons entre mots, la structure de la langue au travers d'un modèle dit de langage. Les domaines d'application des modèles de Markov cachés sont donc naturellement la transcription automatique de la parole [27], mais également l'identification des langues [43, 44], la conversion de voix [45] et la synthèse de parole [46, 47].

Selon le domaine d'application, le type d'observations acoustiques et la topologie du MMC sont les deux choix fondamentaux à prendre. Comme nous

le verrons dans les paragraphes suivants, le cadre spécifique de notre application nous a conduit à privilégier les LPCC comme observations acoustiques auxquelles nous avons adjoint le pourcentage de voisement pour ne gérer qu'un seul MMC. Les paramètres spécifiques au type de trame, par exemple la fréquence fondamentale dans le cas d'une trame voisée, ont été écartés pour traiter tous les types de sons de manière identique. Le processus de génération et les contraintes imposées par la téléphonie n'autorisent pas la prise en compte d'un quelconque modèle de langage ; nous nous sommes limités à une modélisation acoustico-phonétique. Néanmoins, nous avons été amenés à étudier différentes mises en œuvre selon que l'approche est supervisée ou non.

Dès lors que nous recherchons à générer des observations manquantes, lors des pertes de paquets, se pose le problème suivant : pour un modèle de Markov caché donné, comment estime-t-on le vecteur de représentation acoustique $\boldsymbol{\phi}_t, \tau \leq t < \tau + L$ des trames perdues pendant la transmission ? La deuxième partie de ce chapitre propose une solution basée sur la notion de meilleur chemin obtenu à partir des seules données observées.

5.1 Nature du vecteur d'observation

Le modèle de Markov caché modélise l'évolution temporelle d'un vecteur de paramétrisation acoustique du signal, noté $\boldsymbol{\phi}_t$. Ce vecteur, calculé toutes les $10ms$ sur une portion de signal de $30ms$ avec un recouvrement de $20ms$, décrit la trame d'indice t sur laquelle il est calculé.

De nombreux types de paramétrisation du signal de parole ont été largement étudiés, en relation étroite avec le type d'application : les plus anciens sont certainement les coefficients de prédiction linéaire utilisés notamment en codage de parole alors que les coefficients cepstraux se sont montrés performants en reconnaissance de parole, la primauté revenant aux Mel Frequency Cesptral Coefficients (MFCC) [48]. D'autres alternatives peuvent être les Line Spectral Frequency (LSF) [28], moins sensibles à la quantification et

donc largement utilisés dans les codeurs de parole ou encore les Perceptual Linear Pairs (PLP) [49].

D'une part, notre travail a pour vocation d'être utilisé en téléphonie et d'autre part, nous avons besoin de résultats satisfaisants en reconnaissance phonétique ; c'est pourquoi la recherche d'un compromis entre codage et reconnaissance nous a conduit à choisir comme paramétrisation de base les coefficiens cepstraux issus de la prédiction linéaire [24]. Le pourcentage de voisement [37] est adjoint à ce vecteur paramètre pour représenter la composante non-harmonique du signal. La dynamique du signal de parole est prise en compte en introduisant les dérivées premières de ces coefficients pour représenter le lien temporel de tous ces paramètres. Le vecteur d'observation de la trame t est donc composé des 22 coefficients suivants :

- le pourcentage de voisement ($v_{\%}$),
- 10 coefficients cepstraux issus de la prédiction linéaire ($LPCC$),
- la dérivée première du pourcentage de voisement ($\Delta v_{\%}$),
- les dérivées premières des coefficients LPCC ($\Delta LPCC$).

Les dérivées des coefficients sont calculées à l'aide d'une approximation numérique d'ordre 3. Dans le cas de paquets reçus, la dérivée temporelle Δx de la coordonnée x est calculée à l'aide de la formule centrée (5.1) et est appelée "dérivée centrée" :

$$\Delta x_t = \frac{x_{t+1} - x_{t-1}}{2} \tag{5.1}$$

Sur les bords des fenêtres d'observations ou aux bords des trames manquantes, la dérivée est calculée à l'aide des formules décentrées obtenues à partir des développements en série de Taylor. Ces formules sont :

- dérivée à gauche (bord droit) :

$$\Delta x_t = \frac{3x_{t-2} + x_t - 4x_{t-1}}{2} \tag{5.2}$$

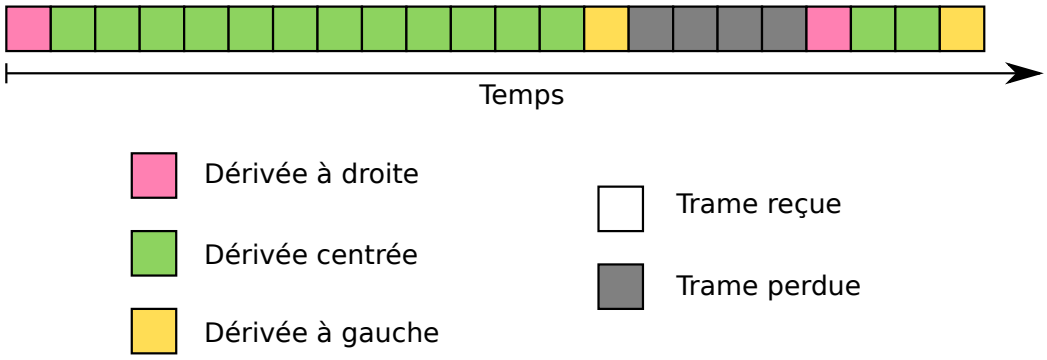


FIGURE 5.1 – Calculs des dérivées des paramètres d'observation lors de phénomènes de bords.

- dérivée à doite (bord gauche) :

$$\Delta x_t = \frac{4x_{t+1} - 3x_t - x_{t+2}}{2} \tag{5.3}$$

La figure 5.1 montre le déroulement du calcul de la dérivée des paramètres sur les bords, notamment lors de pertes de paquets.

5.2 Topologie et estimation des paramètres des modèles de Markov cachés

Nous avons été amenés à aborder deux approches pour résoudre les problèmes de topologie et d'estimation des paramètres des modèles de Markov cachés. Les deux approches s'appuient sur un corpus d'apprentissage pour estimer les paramètres des modèles. Ce corpus conditionne à la fois la langue et le type de parole. Il en résulte que le modèle de Markov caché ainsi appris sera dépendant de la langue. Néanmoins les travaux de Martine Ada et al. [50] sur le traitement automatique multilingue de la parole laissent penser que la production d'un modèle universel est possible. Nous n'avons pas vraiment exploré cette indépendance.

Les deux approches étudiées sont liées à l'ajout ou non de connaissances expertes dans la spécification de la topologie du modèle :

En apprentissage supervisé, le nombre d'états et la structure du modèle sont contraints : seuls certains enchaînements d'états sont permis, les autres sont interdits.

En apprentissage non supervisé, seul le nombre d'états est contraint. Toutes les transitions sont permises : le modèle est dit ergodique.

Les différentes versions des modèles proposés sont comparées.

5.2.1 Approche supervisée

L'approche supervisée, couramment utilisée en reconnaissance de la parole, repose sur une connaissance *a priori* de la linguistique et donc de la langue. Toute phrase se décompose en éléments sonores ; un élément est en général assimilé à un *phone* et est modélisé par un modèle de Markov caché élémentaire. Les différents modèles de Markov élémentaires, estimés indépendamment les uns des autres, sont alors concaténés pour donner le modèle de Markov représentant l'évolution sonore globale propre à cette langue. C'est ce dernier modèle qui par la suite est utilisé pour le masquage de la perte des paquets. Lors de l'étape de concaténation, des règles de prononciation peuvent être prises en compte au travers d'une matrice de transition inter-*phones*.

Les modèles élémentaires

Dans cette approche, chaque *phone* est représenté par un modèle de Markov caché ayant une structure gauche-droite. Chaque modèle comporte trois états correspondant grossièrement au début, au milieu et à la fin du phone. La figure 5.2 présente cette structure.

On note $a_{ij} = \mathrm{P}\left(q_t = j | q_{t-1} = i\right)$. Les états "*start*" et "*end*" représentent respectivement l'entrée et la sortie du phonème. Ainsi,

$$a^{\text{phone}}_{\text{start},j} = \mathrm{P}(q_t = j | q_{t-1} = \text{"start"})$$

représente la probabilité d'entrée sur le modèle considéré (donc dans le *phone*

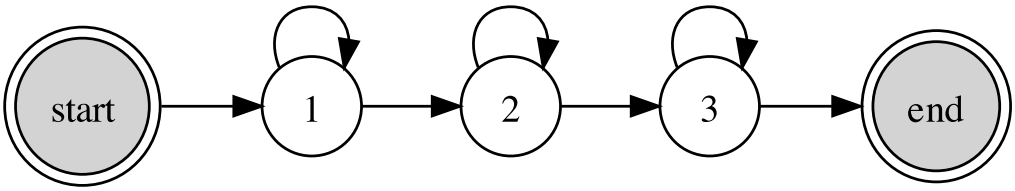


FIGURE 5.2 – Modèle de Markov gauche droite à trois états.

correspondant) sur l'état j. De même,

$$a_{i,\text{end}}^{\text{phone}} \, \text{P}(q_{t+1} = \text{"end"} | q_t = i)$$

représente la probabilité de sortir du *phone* à partir de l'état i. Ces deux états permettent de faciliter les calculs lors de la transition entre les différents phones.

Sur chaque état, la loi des observations est supposée normale. Ainsi pour tous les temps t, q_t est la variable aléatoire représentant l'état à l'instant t, $\text{P}(\boldsymbol{\phi_t} | q_t = i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ où $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ est une loi normale de moyenne $\boldsymbol{\mu}_i$ et de matrice de covariance $\boldsymbol{\Sigma}_i$. Dans le modèle de Markov caché, cette loi ne dépend que de l'état courant et non du temps. Les paramètres de ces différentes lois sont estimés en utilisant un corpus de parole étiqueté phonétiquement (comme par exemple OGI_MLTS ou BREF80BE).

Chaque modèle élémentaire est appris séparément sur la partie dévolue à l'apprentissage du corpus de parole choisi. L'apprentissage se déroule en deux phases :

- l'initialisation à l'aide d'un algorithme de type K-means,
- l'optimisation des paramètres à l'aide de l'algorithme de Baum-Welch.

On obtient ainsi autant de modèles élémentaires que de phones.

Modèle global

Dans notre système de masquage de pertes de paquets, un modèle acoustique global incluant tous les phones est utilisé, il résulte de la concaténation de

tous les modèles phonétiques élémentaires. Cette concaténation est réalisée en tenant compte des différentes probabilités d'enchaînements entre les modèles de phones. Dans notre étude, nous n'avons pas cherché à introduire un modèle phonotactique pour rendre compte de ces enchaînements. Les probabilités de sortir d'un modèle et d'entrer dans un autre sont déduites des probabilités initiales et des probabilités de sortie des différents modèles élémentaires (les transitions à partir de l'état fictif *"start"* et vers l'état fictif *"end"*). Aucun autre apprentissage n'a été effectué. Quelques tentatives n'ont pas démontré leur intérêt.

La matrice de transition du modèle de Markov caché global ainsi obtenue est alors très structurée. Le schéma 5.3 représente une matrice de transition relative à un tel modèle. Les matrices de transition internes à chaque *phone* correspondent aux cases de couleur rose, alors qu'en bleu sont représentées les transitions entre les *phones*. On remarque sur cette figure que cette matrice est assez creuse : la structure du modèle est fortement contrainte par les notions introduites par la linguistique. Elle tient compte des spécificités de la langue à travers les connaissances introduites *a priori* dans les transitions entre les *phones*. Sur la figure 5.4 sont représentées les valeurs de la matrice de transition du modèle global après apprentissage ; la couleur "bleu foncé" est utilisée pour représenter les valeurs nulles des probabilités de transition. Cette matrice a été apprise sur le corpus OGI_MLTS décrit dans le chapitre 3, en introduisant 48 *phones*.

L'annotation acoustique fine en phonèmes de corpora est extrêmement coûteuse en termes de temps et d'expert. L'objectif de cette étude n'est pas de reconnaître les *phones* effectivement prononcés, mais de combler les sons manquants lors de pertes de paquets en téléphonie. Nous proposons donc dans le paragraphe suivant une méthode pour concevoir un modèle de Markov caché global indépendant de toute annotation : c'est l'approche non-supervisée.

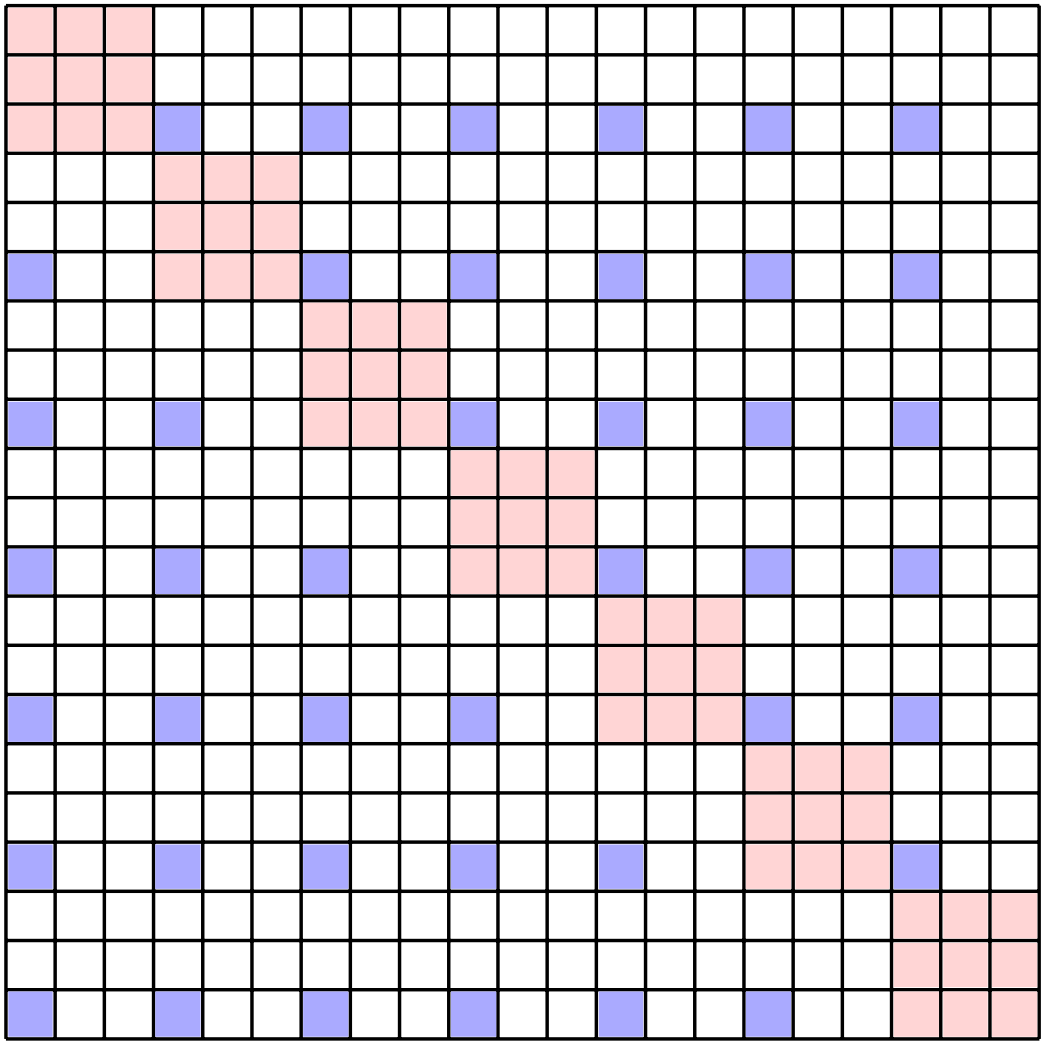


FIGURE 5.3 – Schéma d'une matrice de transition supervisée
Cette matrice comporte sept phones et trois états par phone. En rose les matrices propres à chaque phone. En bleu, les transition entre les phones. En blanc les transitions interdites.

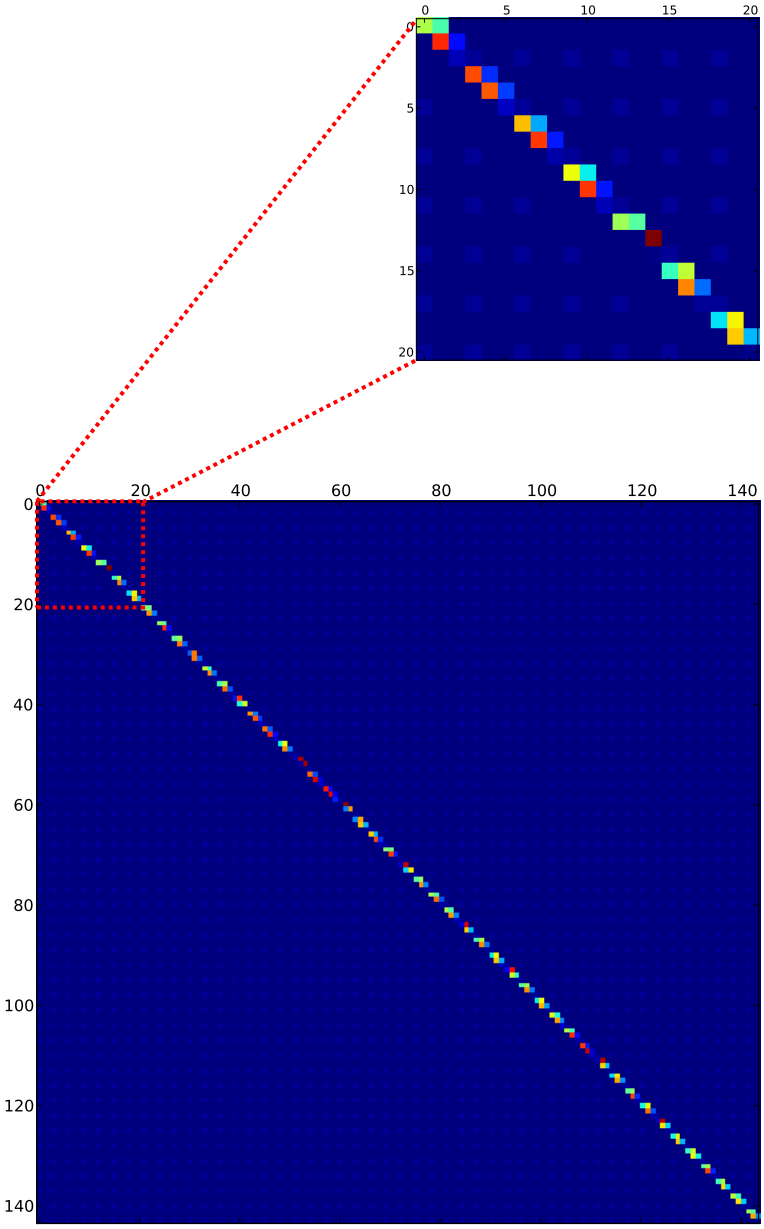


FIGURE 5.4 – Valeurs de la matrice de transition après apprentissage dans le cadre d'une approche supervisée.

Le bleu foncé correspond à des valeurs nulles.

Cette matrice a été apprise sur le corpus OGI_MLTS. Nous avons utilisé 48 *phones* ce qui fait un total de 144 états pour le modèle de Markov caché global.

5.2.2 Approche non-supervisée

Contrairement à l'approche supervisée, en mode non-supervisé, aucune supposition n'est faite sur la structure des modèles, en particulier, elle n'est pas guidée par les spécificités du langage. L'avantage est qu'aucune donnée experte n'est nécessaire afin d'effectuer le réglage du modèle : celui-ci se fait à travers des algorithmes automatiques.

Le nombre d'états Q du modèle de Markov est choisi au préalable. La structure du modèle est supposée ergodique : toutes les transitions entre les états sont possibles et aucune contrainte n'est imposée *a priori* sur l'enchaînement des états.

Le modèle est obtenu en deux étapes :

- Tous les vecteurs d'observations du corpus d'apprentissage sont soumis à un algorithme de type clustering non supervisé. L'algorithme des K-means [51] est utilisé pour regrouper les vecteurs d'observations en Q classes représentant les sonorités de base. Une distribution normale de moyenne $\boldsymbol{\mu}_i$ et de matrice de covariance $\boldsymbol{\Sigma}_i$ est estimée pour modéliser les données de la classe i. Cette loi sert de loi d'observation pour l'état i.

- Du fait du clustering, chaque vecteur d'observation du corpus d'apprentissage est associé à un état et donc labellisé. La matrice des probabilités de transitions $\mathbf{A}$ est obtenue en calculant les occurrences des couples de labels dans le corpus. Ainsi la probabilité $a_{i,j} = \mathrm{P}\left(q_t = j | q_{t-1} = i\right)$ est estimée comme le nombre de fois que le label j est précédé du label i dans le corpus, normalisé par le nombre de labels i :

$$a_{ij} = \frac{\text{Nombre d'occurrences du couple } (i,j)}{\text{Nombre d'occurrences du label } i} \tag{5.4}$$

On constate sur les figures 5.5 et 5.6 que la matrice de transition ainsi obtenue est néanmoins fortement structurée : seul un nombre limité d'états sont connectés entre eux. Cette structure est similaire à celle obtenue avec un apprentissage supervisé où les états inter-phone sont connectés ensemble et seuls les états de sorties et d'entrées sont connectés aux autres phones.

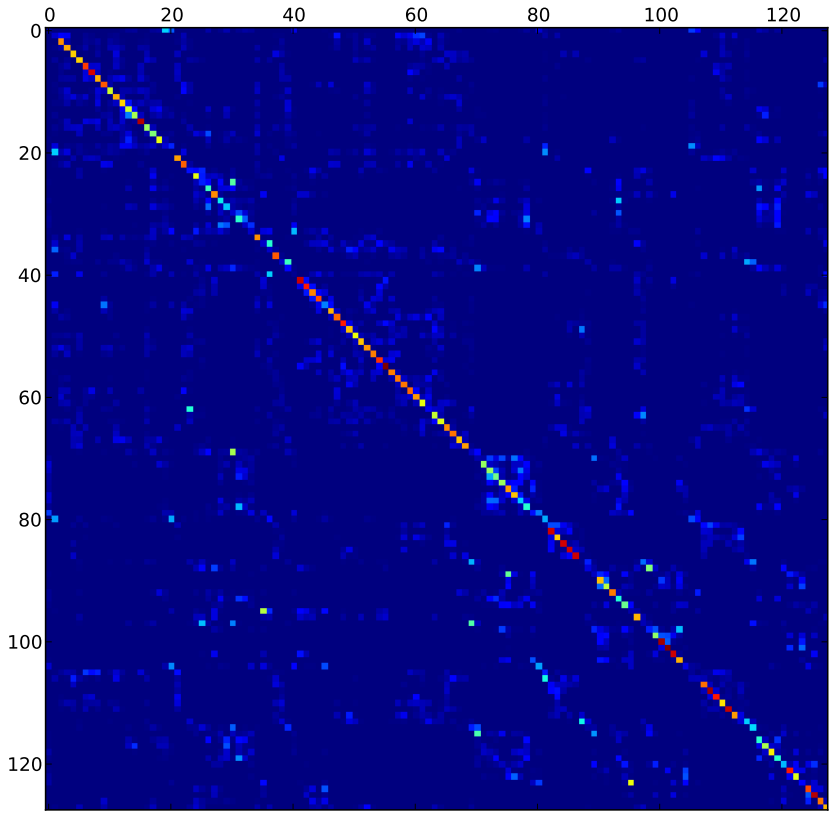


FIGURE 5.5 – Matrice de transition obtenue dans le cadre d'une approche non-supervisée

Cette matrice est apprise sur le corpus BREF80BE bande étroite. Le nombre d'états du modèle de Markov caché est fixé a priori *à 128 états.*

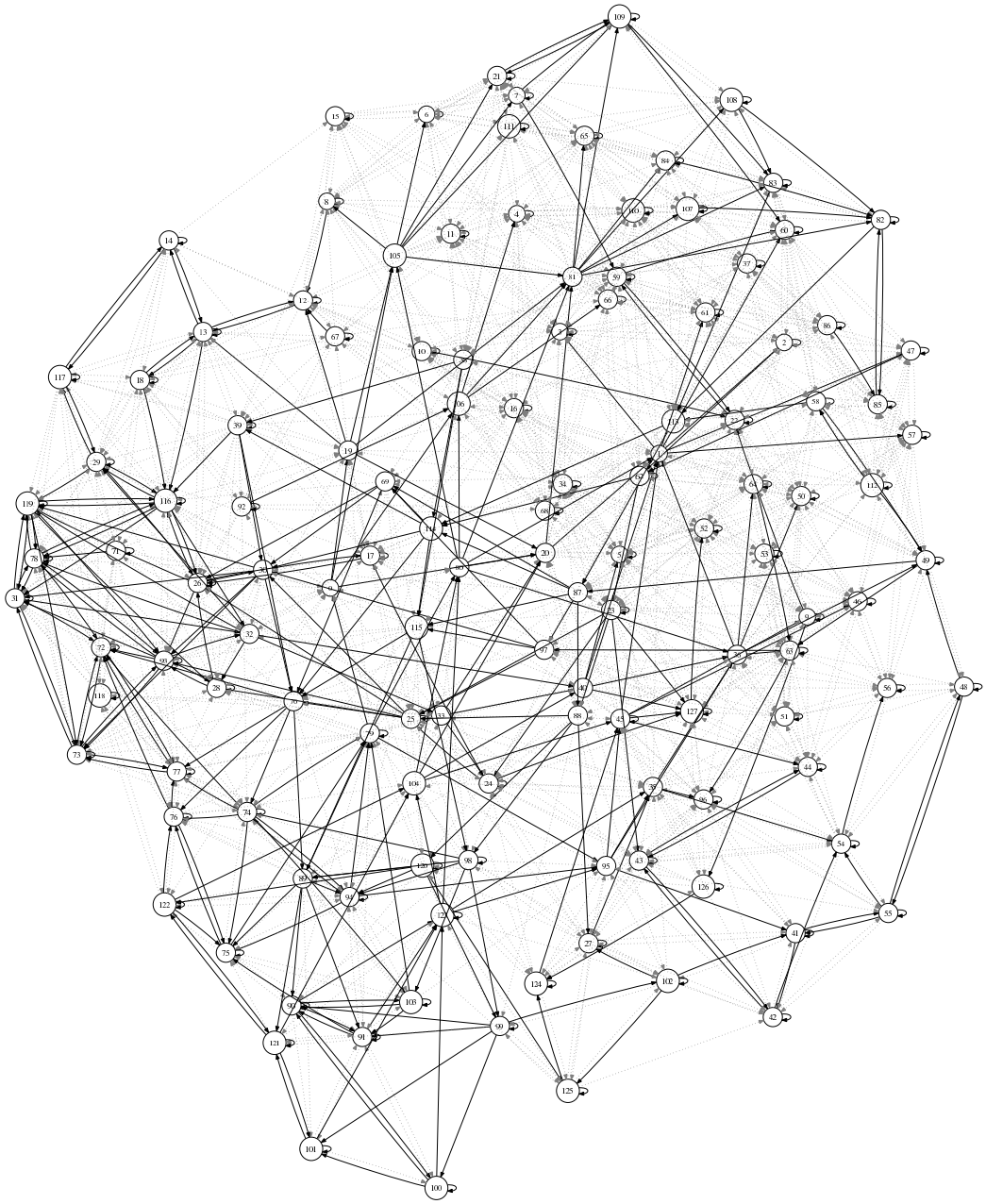


FIGURE 5.6 – Schéma du modèle `bref80be_kmeans128`.
Modèle de Markov caché appris sur le corpus BREF80BE bande étroite. Le nombre d'états du modèle de Markov caché est fixé a priori *à 128 états. Les lignes noires représentent les probabilités de transition supérieures à* 0.06*, les lignes pointillées grises celles comprises entre* 0.02 *et* 0.06*. Les probabilités de transitions inférieures à* 0.02 *ne sont pas tracées.*

5.2.3 Comparaison des modèles

Dans la mesure où nous avons comme objectif d'utiliser ces modèles en mode "génération d'observation", il convient de chercher à les évaluer dans ce contexte.

Nous proposons deux manières de générer des observations en utilisant le modèle de Markov caché appris. Ces deux approches seront validées en les appliquant sur des corpora de parole originaux, sans perte de paquets, et en comparant les observations générées avec les valeurs réelles.

Les deux approches proposées pour la génération d'observations sont basées sur :

- l'utilisation d'un mélange de lois gaussiennes (GMM),
- ou sur l'utilisation du meilleur chemin déterminé à partir de l'algorithme de Viterbi.

Ces deux approches sont détaillées ci-après.

Génération par mélange de lois gaussiennes (GMM)

Dans cette première approche, la génération d'observations est obtenue en s'inspirant des travaux de Rødbro [25] : l'observation générée à l'instant t, connaissant toute la suite d'observations réelles, suit une loi de type mélange de lois gaussiennes donnée par :

$$\boldsymbol{\phi}_t^{\mathrm{GMM}} \sim \sum_{i=1}^{Q} \mathrm{P}\left(q_t = i | \boldsymbol{\phi}_1 \cdots \boldsymbol{\phi}_{\tau+L+J-1}\right) b_i \tag{5.5}$$

où b_i est la loi des observations, loi gaussienne de moyenne $\boldsymbol{\mu}_i$, correspondant au i^{e} état du modèle de Markov. En reprenant les notations classiques utilisées dans la littérature [24], cette observation s'écrit :

$$\tilde{\boldsymbol{\phi}}_t^{\mathrm{GMM}} = \sum_{i=1}^{Q} \gamma_t(i) \boldsymbol{\mu}_i \tag{5.6}$$

où $\gamma_t(i)$ est définie de la manière suivante :

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{Q} \alpha_t(j)\beta_t(j)} \tag{5.7}$$

avec

$$\begin{aligned}\alpha_t(i) &= \mathrm{P}\left(q_t = i, \boldsymbol{\phi}_1^t\right) \\ \beta_t(i) &= \mathrm{P}\left(\boldsymbol{\phi}_{t+1}^{\tau+L+J-1} | q_t = i\right)\end{aligned}$$

.

Suivant la façon d'estimer les paramètres du modèle de Markov caché, la génération par GMM (5.6) a plus ou moins de sens. En effet, si le modèle de Markov caché est appris de manière supervisée, comme présenté au paragraphe 5.2.1, la génération (5.6) va conduire à "inventer" un *phone* non prévu dans l'analyse préalable en construisant une combinaison linéaire d'états appartenant à différents *phones*. Ceci va à l'encontre de l'approche supervisée qui prévoit une structure très précise du modèle. De plus, en construisant le modèle de Markov caché de manière non-supervisée, on s'aperçoit que l'estimation des paramètres conduit tout de même à un modèle fortement structuré comme l'illustre la figure 5.5, avec une structure similaire à celle de l'approche supervisée.

Cette remarque nous amène à simplifier la génération (5.6) afin de lui redonner un sens au niveau phonétique. Nous proposons de remplacer la somme par le terme prédominant correspondant à l'état le plus probable à l'instant t :

$$\tilde{\boldsymbol{\phi}}_t^{\mathrm{GMMBS}} = \boldsymbol{\mu}_{\tilde{q}_t} \tag{5.8}$$

$$\tilde{q}_t = \underset{i=1,\cdots,Q}{\operatorname{argmax}}\, \gamma_t(i) \tag{5.9}$$

Génération par recherche du meilleur chemin (Viterbi)

Une autre possibilité que nous avons exploitée par la suite est de générer des observations en nous appuyant sur le meilleur chemin ayant généré les observations connues. C'est pourquoi nous proposons également d'évaluer nos modèles en comparant l'observation réelle et l'observation générée à partir de ce meilleur chemin, à savoir :

$$\tilde{\boldsymbol{\phi}}_t^{\text{Viterbi}} = \boldsymbol{\mu}_{q_t^\star} \tag{5.10}$$

où $q_t^\star$ est l'état atteint à l'instant t au cours du meilleur chemin. Ce chemin est trouvé à l'aide de l'algorithme de Viterbi [24].

Résultats

Nous avons évalué chacun des modèles (approches supervisée et non-supervisée) en comparant l'observation réelle aux observations générées selon les deux méthodes, observations notées respectivement $\tilde{\boldsymbol{\phi}}^{\text{GMM}}$ et $\tilde{\boldsymbol{\phi}}_t^{\text{Viterbi}}$, et ce par distance euclidienne entre les vecteurs.

Les modèles de Markov cachés évalués varient selon le mode de l'approche supervisée/non-supervisée, selon le type des lois gaussiennes (matrice diagonale ou non), selon le nombre d'états et selon le corpus d'apprentissage.

Dans un souci de lisibilité, des conventions d'écriture pour le nom des modèles ont été adoptées pour refléter les différentes combinaisons expérimentales étudiées. C'est ainsi que le préfixe du nom du modèle stipule le corpus ayant servi à l'apprentissage, *bref80be* pour BREF80BE et *ogi_mlts* pour le corpus OGI, alors que le suffixe précise la méthode d'apprentissage.

Les modèles dont le suffixe est de la forme *kmeansXX* sont des modèles comportant XX états appris de manière non-supervisée. Les lois d'observations sont alors supposées normales avec une matrice de covariance pleine.

Les modèles ayant comme suffixe *diag* sont des modèles appris selon la méthode supervisée, en prenant trois états par *phone* et une loi d'observation normale dont la matrice de covariance est diagonale.

Finalement, les modèles suffixés *plain* sont également construits à l'aide de la méthode supervisée, nous avons pris trois états par *phone*, la loi d'observation est normale mais la matrice de covariance est quant à elle pleine.

Dans la suite de notre étude, nous proposons de comparer les modèles suivants :

bref80be_kmeans32 : ce modèle inclut $Q = 32$ états issus d'un apprentissage non-supervisé. Les lois d'observations sont supposées gaussiennes avec des matrices de covariances pleines.

bref80be_kmeans128 : obtenu de la même manière que **bref80be_kmeans32**, seul le nombre d'états est plus important : $Q = 128$. Remarquons que le nombre d'états est alors du même ordre de grandeur que celui utilisé par Rødbro dans [25].

bref80be_kmeans512 : identique aux précédents modèles, le nombre d'états est maximum : $Q = 512$.

ogi_mlts_plain : ce modèle est un modèle appris de manière supervisée sur la partie annotée phonétiquement corpus OGI_MLTS en langue *anglaise*. Il comporte $Q = 144$ états modélisant les 48 phonèmes de l'anglais présents et étiquetés dans ce corpus. Les lois d'observations sont supposées gaussiennes avec des matrices de covariances pleines.

ogi_mlts_diag : ce modèle se distingue du précédent dans le sens où les matrices de covariances sont diagonales.

bref80be_plain : ce modèle à $Q = 105$ états est appris sur le corpus BREF80BE de manière supervisée. Chaque modèle élémentaire est composé de trois états. Le modèle global modélise ainsi les 35 phonèmes du français. Les lois d'observations sont des lois gaussiennes avec des matrices de covariances pleines.

bref80be_diag : Identique dans sa structure au modèle précédent, celui-ci se distingue par ses lois d'observations ; en effet, celles-ci sont supposées gaussiennes, mais avec des matrices de covariances diagonales.

Il est rappelé que le corpus de test utilisé est le sous-ensemble de test de BREF80BE. Des différences importantes entre les conditions d'apprentissage et de test (environnement, langue), sont à noter ; ce "mismatch" devrait se traduire par des distances plus importantes lorsque l'apprentissage a été fait sur OGI_MLTS.

Dans le tableau 5.1, est donnée la valeur moyenne des distances entre observations générées, selon leur type de génération, et observations réelles, ainsi que l'écart-type correspondant.

Nous pourrions nous interroger sur la pertinence d'un calcul de distance euclidienne sur un vecteur non-homogène constitué de paramètres divers (cf. paragraphe 5.1). Toutefois, il faut remarquer que la génération par GMM a été faite de façon à minimiser l'erreur quadratique sur ce vecteur composite et que la génération par Viterbi est réalisée de manière à maximiser la vraisemblance du chemin sachant la suite des vecteurs composites.

Il est clair que l'approximation la meilleure est obtenue en générant une observation avec l'approche GMM. Cependant si l'on compare les distances moyennes pour le générateur Viterbi et les modèles `bref80be_diag` et `ogi_mlts_diag`, la modélisation semble indépendante de la langue du corpus ayant servi à l'apprentissage. Cette remarque permet d'envisager, comme il a été mentionné précédemment, une extension en recherchant un modèle acoustique universel.

Afin de vérifier que le couple (moyenne, écart-type) a un sens et est correctement interprété, nous traçons sur la figure 5.7 quelques-unes des distributions des distances. La plupart de ces distributions s'apparentent à des distributions gaussiennes dont les moyennes et les écarts-types sont précisés sur chaque histogramme. Ce qui nous intéresse ici, c'est que les lois sont "piquées" autour de la valeur moyenne, ce qui donne du sens à l'interprétation des valeurs moyennes. Cet examen confirme le bon comportement de l'approche GMM, sans toutefois rejeter l'approche Viterbi puisque l'intervalle (moyenne/écart-type, moyenne + écart-type) reste significatif.

Le figure 5.8 apporte un éclairage complémentaire en présentant la dis-

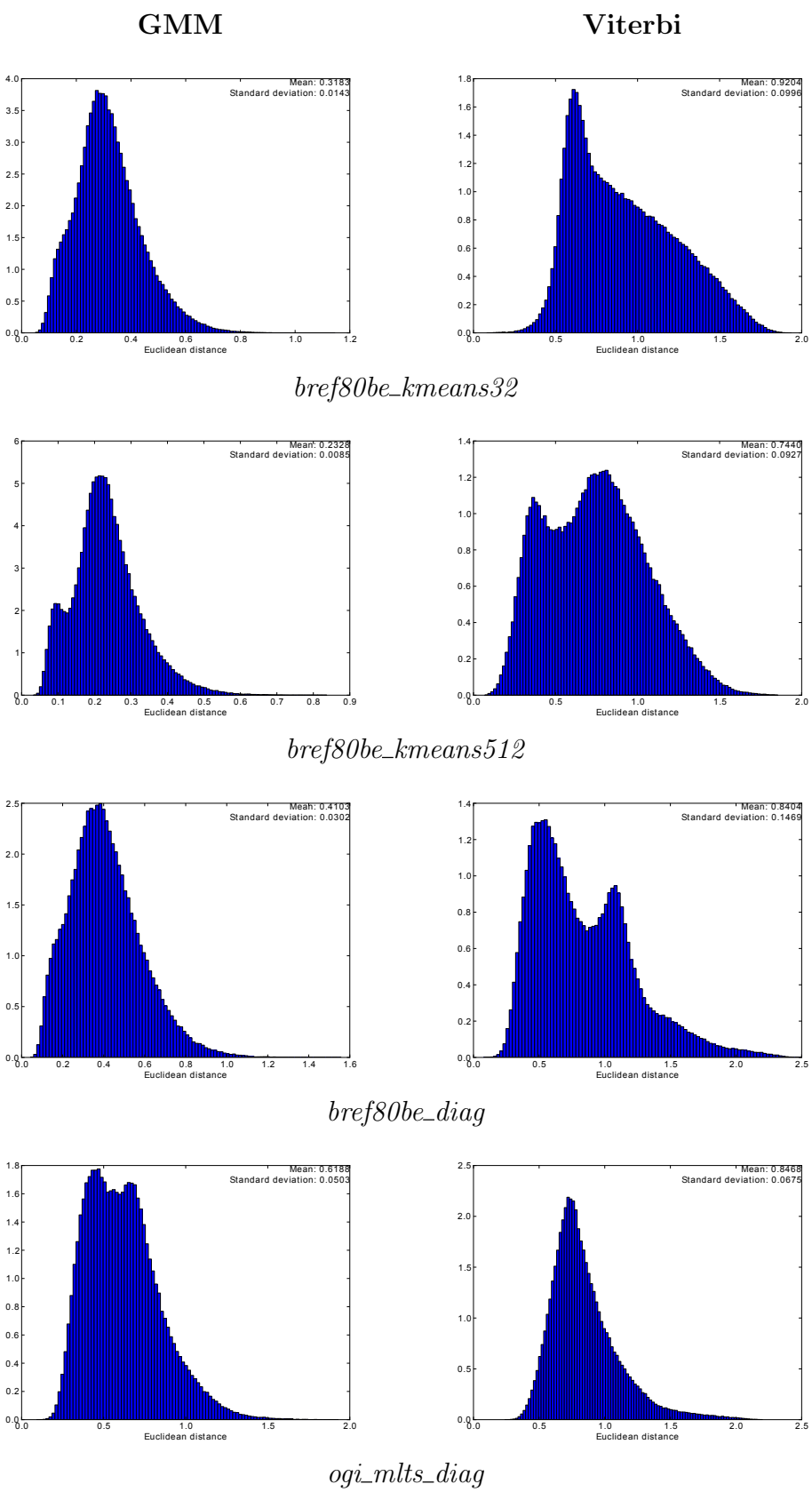


FIGURE 5.7 – Distributions des distances entre les observations et les observations estimées selon le type de génération.

Modèle	GMM		Viterbi	
	Moyenne	Écart-type	Moyenne	Écart-type
bref80be_kmeans32	0.31	0.01	0.92	0.09
bref80be_kmeans128	0.26	0.01	0.92	0.14
bref80be_kmeans512	0.23	0.008	0.74	0.09
bref80be_diag	0.41	0.03	0.84	0.14
bref80be_plain	0.42	0.03	0.72	0.06
ogi_mlts_diag	0.61	0.05	0.84	0.06
ogi_mlts_plain	0.62	0.06	1.11	0.22

TABLE 5.1 – Erreur moyenne de modélisation.

tance moyenne pour chacune des composantes du vecteur d'observation à savoir :

- le pourcentage de voisement : $v_{\%}$,
- 10 coefficients LPCC,
- la dérivée du pourcentage de voisement : $\Delta v_{\%}$,
- la dérivée des 10 coefficients LPCC : ΔLPCC

Comme nous nous y attendions, les deux modèles appris sur le corpus OGI_MLTS, `ogi_mlts_plain` en bleu et `ogi_mlts_diag` en mauve, sont les moins performants (surface la plus importante) pour ce qui est de l'approche GMM. Nous remarquons également que ce n'est pas le cas pour l'approche Viterbi. En effet, le modèle `ogi_mlts_diag` a des performances du même ordre que `bref80be_plain` lorsque l'approche Viterbi est considérée. On peut noter que l'approche GMM, au contraire de l'approche Viterbi, ne semble pas tenir compte du pourcentage de voisement : pour tous les modèles, la distance moyenne entre les observations et les observations estimées par l'approche GMM est de l'ordre de 0.6.

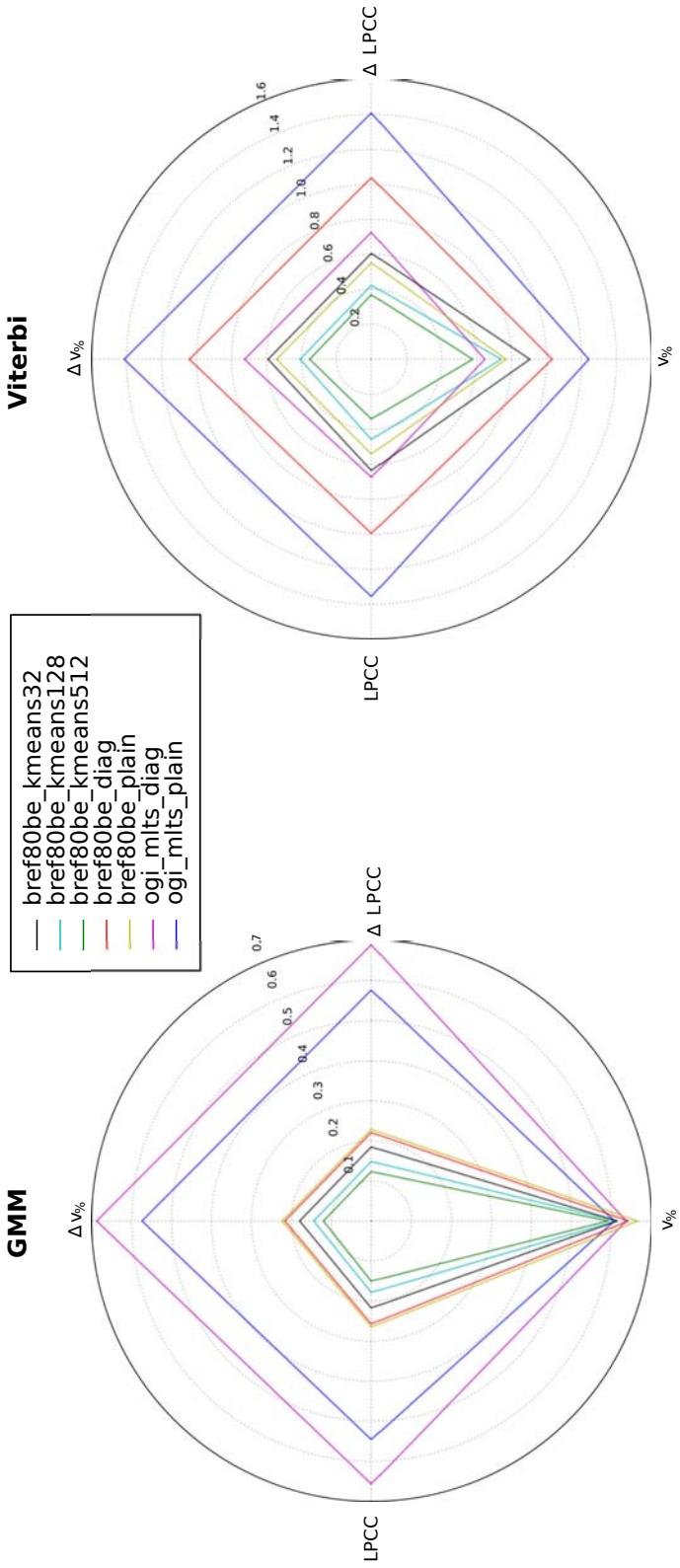


FIGURE 5.8 – Distances moyennes entre observations et observations estimées pour chaque sous-composante du vecteur d'observation.

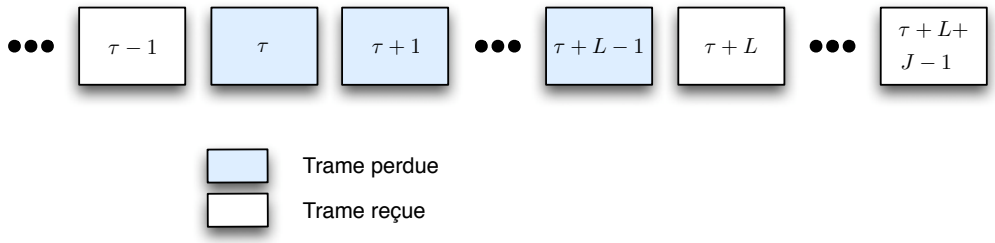


FIGURE 5.9 – États de la suite de paquets.

5.3 Estimation sur le meilleur chemin

Comme nous l'avons vu dans le chapitre 2, paragraphe 2.3, Christoffer Rødbro estime les observations perdues $\boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1}$ à l'aide d'un mélange de lois Gaussiennes. Dans notre étude, nous proposons deux approches différentes pour estimer les observations perdues. Ces deux approches ont été présentées dans la section précédente : une approche inspirée de Rødbro, basée sur des mélanges de lois gaussiennes et une approche liée à l'estimation du meilleur chemin par l'algorithme de Viterbi. Toutefois, cette dernière approche nécessite une étude particulière car l'algorithme classique de Viterbi ne prend pas en compte d'éventuelles données manquantes. Le développement théorique que nous présentons ici conduit à une version modifiée de l'algorithme de Viterbi pour estimer conjointement le meilleur chemin et les observations maximisant ce chemin lors des pertes de paquets.

5.3.1 Rappel des notations

Nous rappelons les notations utilisées. Nous supposons que les trames acoustiques de 1 à $\tau - 1$ et de $\tau + L$ à $\tau + L + J - 1$ sont reçues (cf. figure 5.9).

Pour chaque trame de signal audio reçue, un vecteur de paramétrisation acoustique $\boldsymbol{\phi}_t$ (pour $1 \leq t < \tau, \quad \tau + L \leq t < \tau + L + J$) représentant cette trame est calculé. Nous cherchons à déterminer une estimation $\hat{\boldsymbol{\phi}}_t$ de ce vecteur d'observation lorsque le paquet est perdu, c'est-à-dire lorsque

$\tau \leq t < \tau + L$.

Nous nous appuyons sur une modélisation de l'espace acoustique à l'aide d'un modèle de Markov tel que présenté dans le début de ce chapitre. Q est le nombre d'états du modèle de Markov caché et pour chaque état i, la loi d'observation est notée b_i. La matrice de transition de ce modèle est notée $\boldsymbol{A}$. Ainsi, $a_{ij} = \mathrm{P}\left(q_t = j | q_{t-1} = i\right)$ où q_t est la variable aléatoire représentant le numéro de l'état à l'instant t. De plus, notons $q_{t_1}^{t_2}$ la suite des états $q_{t_1} \cdots q_{t_2}$ et $\boldsymbol{\phi}_{t_1}^{t_2}$ la suite des vecteurs d'observations $\boldsymbol{\phi}_{t_1} \cdots \boldsymbol{\phi}_{t_2}$.

5.3.2 Algorithme de Viterbi lors de pertes de paquets

Nous avons adopté la démarche proposée par le tutoriel de Lawrence Rabiner [27] sur les modèles de Markov cachés, en l'adaptant au cas où certaines observations sont manquantes.

L'expression suivante définit la probabilité du meilleur chemin à partir d'une suite d'observations.

$$\max_{q_1 \cdots q_{\tau+L+J-1}} \mathrm{P}\left[q_1^{\tau+L+J-1}, \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right] \tag{5.11}$$

Dans la mesure où la séquence d'observations $\boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1}$ est manquante, nous sommes amenés à rechercher simultanément les observations manquantes et ce meilleur chemin au sens du maximum de vraisemblance ; nous maximisons conjointement sur les observations manquantes et sur le chemin la probabilité de la suite d'observations totales et du chemin associé.

$$P^\star = \max_{\substack{\boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1} \\ q_1 \cdots q_{\tau+L+J-1}}} \mathrm{P}\left[q_1^{\tau+L+J-1}, \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right] \tag{5.12}$$

En d'autres termes, si l'on pose pour chaque état i

$$\rho_{\tau+L+J-1}(i) = \max_{\substack{\boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1} \\ q_1 \cdots q_{\tau+L+J-1}}} \mathrm{P}\left[q_1^{\tau+L+J-2}, q_{\tau+L+J-1} = i, \boldsymbol{\phi}_1 \cdots \boldsymbol{\phi}_{\tau+L+J-1}\right] \tag{5.13}$$

la probabilité maximum recherchée est donnée par

$$P^\star = \max_{i \in [\![1,Q]\!]} \rho_{\tau+L+J-1}(i)$$

Or en écrivant $\rho_\tau(i)$ en fonction de l'observation manquante $\boldsymbol{\phi}_\tau$:

$$\rho_\tau(i) = \max_{\boldsymbol{\phi}_\tau, q_1 \cdots q_{\tau-1}} \mathrm{P}\left[q_1^{\tau-1}, q_\tau = i, \boldsymbol{\phi}_1^{\tau-1}\boldsymbol{\phi}_\tau\right] \tag{5.14}$$

$$= \max_{\boldsymbol{\phi}_\tau, q_1 \cdots q_{\tau-1}} \mathrm{P}\left[\boldsymbol{\phi}_\tau | q_\tau = i\right] \cdot \mathrm{P}\left[q_\tau = i |\ q_1^{\tau-1}, \boldsymbol{\phi}_1^{\tau-1}\right] \cdot \mathrm{P}\left[q_1^{\tau-1}, \boldsymbol{\phi}_1^{\tau-1}\right] \tag{5.15}$$

$$= \left(\max_{\boldsymbol{\phi}_\tau} \mathrm{P}\left[\boldsymbol{\phi}_\tau | q_\tau = i\right]\right) \max_{q_1 \cdots q_{\tau-1}} \left(\mathrm{P}\left[q_\tau = i |\ q_1^{\tau-1}, \boldsymbol{\phi}_1^{\tau-1}\right] \cdot \mathrm{P}\left[q_1^{\tau-1}, \boldsymbol{\phi}_1^{\tau-1}\right]\right) \tag{5.16}$$

On effectue ensuite le même développement pour le second paquet perdu :

$$\rho_{\tau+1}(j) = \max_{\boldsymbol{\phi}_{\tau+1}\boldsymbol{\phi}_\tau, q_1 \cdots q_\tau} \mathrm{P}\left[q_1^\tau q_{\tau+1} = j, \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_\tau, \boldsymbol{\phi}_{\tau+1}\right] \tag{5.17}$$

$$= \max_{\boldsymbol{\phi}_{\tau+1}\boldsymbol{\phi}_\tau, q_1 \cdots q_\tau} \left(\mathrm{P}\left[\boldsymbol{\phi}_{\tau+1} | q_{\tau+1} = j\right] \cdot \mathrm{P}\left[q_{\tau+1} = j | q_1^\tau, \boldsymbol{\phi}_1^\tau\right] \mathrm{P}\left[q_1^\tau, \boldsymbol{\phi}_1^\tau\right]\right) \tag{5.18}$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+1}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+1} | q_{\tau+1} = j\right]\right) \max_{\boldsymbol{\phi}_\tau, q_1 \cdots q_\tau} \left(\mathrm{P}\left[q_{\tau+1} = j |\ q_1^\tau, \boldsymbol{\phi}_1^\tau\right] \cdot \mathrm{P}\left[q_1^\tau, \boldsymbol{\phi}_1^\tau\right]\right) \tag{5.19}$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+1}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+1} | q_{\tau+1} = j\right]\right) \left(\max_i \left(\underbrace{\mathrm{P}\left[q_{\tau+1} = j | q_\tau = i\right]}_{a_{ij}} \cdot \underbrace{\max_{\boldsymbol{\phi}_\tau, q_1 \cdots q_{\tau-1}} \mathrm{P}\left[q_1^{\tau-1}, q_\tau = i, \boldsymbol{\phi}_1^\tau\right]}_{\rho_\tau(i)}\right)\right)$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+1}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+1} | q_{\tau+1} = j\right]\right) \max_i \left[a_{ij}\rho_\tau(i)\right] \tag{5.20}$$

De même, pour tout $0 \leq k < L$ on peut écrire :

$$\rho_{\tau+k}(j) = \max_{\substack{\boldsymbol{\phi}_{\tau+k}\cdots\boldsymbol{\phi}_\tau \\ q_1\cdots q_{\tau+k-1}}} \mathrm{P}\left[q_1^{\tau+k-1}, q_{\tau+k}=j, \boldsymbol{\phi}_1^{\tau+k-1}, \boldsymbol{\phi}_{\tau+k}\right] \tag{5.21}$$

$$= \max_{\substack{\boldsymbol{\phi}_{\tau+k}\cdots\boldsymbol{\phi}_\tau \\ q_1\cdots q_{\tau+k-1}}} \Bigg(\mathrm{P}\left[\boldsymbol{\phi}_{\tau+k}|q_{\tau+k}=j\right] \cdot \mathrm{P}\left[q_{\tau+k}=j|q_1^{\tau+k-1}, \boldsymbol{\phi}_1^{\tau+k-1}\right] \cdot \mathrm{P}\left[q_1^{\tau+k-1}, \boldsymbol{\phi}_1^{\tau+k}\right] \Bigg) \tag{5.22}$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+k}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+k-1}|q_{\tau+k}=j\right] \right) \cdot \max_{\substack{\boldsymbol{\phi}_{\tau+k-1}\cdots\boldsymbol{\phi}_\tau \\ q_1\cdots q_{\tau+k-1}}} \left(\mathrm{P}\left[q_{\tau+k}=j|q_1^{\tau+k-1}, \boldsymbol{\phi}_1^{\tau+k-1}\right] \cdot \mathrm{P}\left[q_1^{\tau+k-1}, \boldsymbol{\phi}_1^{\tau+k-1}\right] \right) \tag{5.23}$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+k}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+k}|q_{\tau+k}=j\right] \right) \cdot \left(\max_i \Bigg(\underbrace{\mathrm{P}\left[q_{\tau+k}=j|q_{\tau+k-1}=i\right]}_{a_{ij}} \cdot \underbrace{\max_{\substack{\boldsymbol{\phi}_{\tau+k-1} \\ q_1\cdots q_{\tau+k-2}}} \mathrm{P}\left[q_1^{\tau+k-2}, q_{\tau+k-1}=i, \boldsymbol{\phi}_1^{\tau+k-1}\right]}_{\rho_{\tau+k-1}(i)} \Bigg) \right) \tag{5.24}$$

$$= \left(\max_{\boldsymbol{\phi}_{\tau+k}} \mathrm{P}\left[\boldsymbol{\phi}_{\tau+k}|q_{\tau+k}=j\right] \right) \cdot \max_i \left[a_{ij}\rho_{\tau+k-1}(i) \right] \tag{5.25}$$

On en déduit par récurrence la quantité définie dans (5.13).

5.3.3 Estimation des observations manquantes

Nous avons vu précédemment qu'il était nécessaire de déterminer, pour chaque état i, le vecteur d'observation $\hat{\boldsymbol{\phi}}_t^{\,i}$ qui maximise la probabilité d'observation :

$$\hat{\boldsymbol{\phi}}_t^{\,i} = \operatorname*{argmax}_{\boldsymbol{\phi}_t} \mathrm{P}\left[\boldsymbol{\phi}_t|q_t=i\right] \tag{5.26}$$

Dans le cas où la loi des observations de l'état i, b_i, est une loi gaussienne, c'est-à-dire $b_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ pour tout i tel que $1 \leq i \leq Q$, l'observation qui

maximise cette loi est la moyenne $\boldsymbol{\mu}_i$ de la loi. En effet :

$$\begin{aligned}\hat{\boldsymbol{\phi}}_t^{\,i} =& \underset{\boldsymbol{x}}{\operatorname{argmax}} \frac{1}{\sqrt{(2\pi)^D \det(\boldsymbol{\Sigma}_i)}} \exp\left((\boldsymbol{x}-\boldsymbol{\mu}_i)^\dagger \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_i)\right) \\ =& \boldsymbol{\mu}_i \end{aligned} \tag{5.27}$$

où D est la taille du vecteur d'observation. Nous remarquons alors que $\hat{\boldsymbol{\phi}}_t^{\,i}$ ne dépend pas de l'instant considéré mais uniquement de l'état. Dans la suite, on note :

$$\hat{\boldsymbol{\phi}}_t^{\,i} = \hat{\boldsymbol{\phi}}^{\,i}$$

Perspectives : dans le cas d'un mélange de plusieurs lois gaussiennes, la détermination de l'observation maximisant la probabilité d'observation est plus ardue. En effet, il faut résoudre le système (5.28) ci-après dans lequel M est le nombre de lois gaussiennes dans le mélange, $\boldsymbol{\mu}_i^{(l)}$ et $\boldsymbol{\Sigma}_i^{(l)}$ sont respectivement la moyenne et la matrice de covariance de la l^{e} composante gaussienne de la loi des observations de l'état i.

$$\hat{\boldsymbol{\phi}}^{\,i} = \underset{x\in R^D}{\operatorname{argmax}} \sum_{l=0}^{M-1} w_l^i \frac{1}{\sqrt{(2\pi)^D \det\left(\boldsymbol{\Sigma}_i^{(l)}\right)}} \exp\left(\left(\boldsymbol{x}-\boldsymbol{\mu}_i^{(l)}\right)^\dagger \left(\boldsymbol{\Sigma}_i^{(l)}\right)^{-1} \left(\boldsymbol{x}-\boldsymbol{\mu}_i^{(l)}\right)\right) \tag{5.28}$$

où D est la dimension du vecteur d'observation, w_l^i est le poids de la l^{e} gaussienne dans la loi d'observation de l'état i.

La résolution d'un tel système n'est pas aisée et on doit faire appel à un algorithme d'optimisation. L'approche multigaussienne n'ayant pas été utilisée dans le cadre de cette thèse, les développements n'ont pas été effectués.

5.3.4 Algorithme de Viterbi modifié

"L'algorithme de Viterbi avec des observations manquantes" s'écrit par conséquent :

Initialisation Cette initialisation est identique à celle de l'algorithme de Viterbi sans perte d'observations, nous supposons que le premier paquet n'est pas perdu. Ainsi pour tout état j compris entre 1 et Q :

$$\begin{aligned}\rho_0(j) =& \pi_j \, \mathrm{P}\left[\boldsymbol{\phi}_0 | q_0 = j\right] \\ \xi_0(j) =& 0\end{aligned}$$

où π_j est la probabilité initiale d'être dans l'état j.

Propagation À chaque instant $t > 0$, la mise à jour des paramètres se fait de la manière suivante :

- Dans le cas où l'observation $\boldsymbol{\phi}_t$ est disponible, c'est-à-dire pour $t < \tau$ et $\tau + L \leq t < \tau + L + J$: pour tout j compris entre 1 et Q :

$$\rho_t(j) = \max_i \rho_{t-1}(i) a_{ij} \, \mathrm{P}\left[\boldsymbol{\phi}_t | q_t = j\right] \tag{5.29}$$

$$\xi_t(j) = \operatorname*{argmax}_i \rho_{t-1}(i) a_{ij} \, \mathrm{P}\left[\boldsymbol{\phi}_t | q_t = j\right] \tag{5.30}$$

- Dans le cas où l'observation $\boldsymbol{\phi}_t$ n'est pas disponible, c'est-à-dire pour $\tau \leq t < \tau + L$: pour tout j compris entre 1 et Q :

$$\rho_t(j) = \max_i \rho_{t-1}(i) a_{ij} \max_{\boldsymbol{\phi}_t} \mathrm{P}\left[\boldsymbol{\phi}_t | q_t = j\right] \tag{5.31}$$

$$\xi_t(j) = \operatorname*{argmax}_i \rho_{t-1}(i) a_{ij} \tag{5.32}$$

$$\hat{\boldsymbol{\phi}}_t^j = \operatorname*{argmax}_{\boldsymbol{\phi}_t} \mathrm{P}\left[\boldsymbol{\phi}_t | q_t = j\right] \tag{5.33}$$

  Nous remarquons que la loi des observations est indépendante du temps et ne dépend que de l'état. Ainsi $\hat{\boldsymbol{\phi}}_t^j$ est indépendant du temps. On peut donc écrire : $\hat{\boldsymbol{\phi}}_t^j = \hat{\boldsymbol{\phi}}^j$.

Fin

$$P^\star = \max_i \rho_{\tau+L+J}(i) a_{ij} \, \mathrm{P}\left[\boldsymbol{\phi}_{\tau+L+J} | q_{\tau+L+J} = j\right] \tag{5.34}$$

$$q^\star_{\tau+L+J} = \operatorname*{argmax}_i \rho_{\tau+L+J}(i) \tag{5.35}$$

Rétro-propagation Le meilleur chemin est obtenu de la manière suivante :

$$q_t^\star = \xi_{t+1}(q_{t+1}^\star) \qquad t = \tau + L + J - 1, \tau + L + J - 2, \cdots, 1 \tag{5.36}$$

$$\hat{\boldsymbol{\phi}}_t = \hat{\boldsymbol{\phi}}_t^{q_t^\star} \qquad t = \tau, \tau + 1, \cdots, \tau + L - 1 \tag{5.37}$$

Chapitre 6

Le système de masquage de pertes de paquets

Nous avons étudié dans les chapitres précédents les différentes briques à la mise en œuvre du système de masquage de pertes de paquets que nous proposons à savoir :

- un nouveau paramètre continu de mesure d'harmonicité (le pourcentage de voisement) ;
- une revisite de l'approche non-supervisée d'estimation des paramètres d'un modèle de Markov caché acoustique ;
- ainsi qu'une nouvelle approche pour l'estimation des observations dans le cas d'observations manquantes.

Après avoir décrit le contexte applicatif et ses contraintes, les systèmes de masquage de pertes de paquets que nous avons proposés sont présentés en deuxième partie de ce chapitre. Les variantes principales sont liées au mode de réestimation des paquets manquants ; nous avons retenu les trois approches que sont :

- l'estimation par mélanges de lois gaussiennes (estimation proposée par Rødbro),

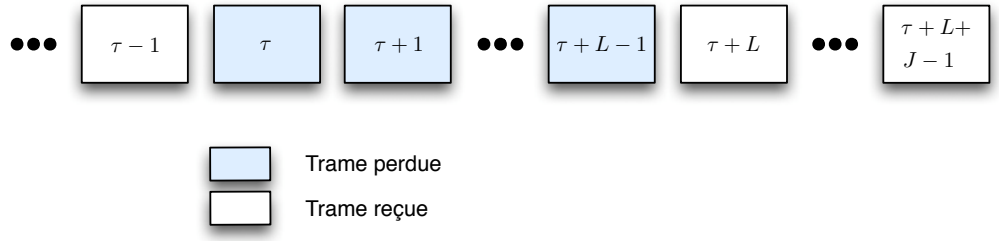


FIGURE 6.1 – État des paquets.

- l'estimation approchée en limitant le nombre de composantes gaussiennes,
- l'estimation originale par l'algorithme de Viterbi modifié.

Les différentes propositions sont évaluées et comparées dans la troisième partie. Les performances des systèmes sont vérifiées de manière objective en s'appuyant sur une distance entre observation estimée et observation cible à l'image de ce qui a été fait au chapitre précédent pour évaluer les modèles de Markov cachés. Puis une évaluation de la qualité du signal de parole produit par le système global est présentée.

6.1 Contexte applicatif

Comme nous l'avons introduit dans le chapitre 1, un certain nombre de paquets de parole peuvent être disponibles en avance dans le tampon de compensation de gigue. Ce phénomène provient des multiples trajets qu'est susceptible d'emprunter chaque paquet. Le tampon de compensation de gigue a pour rôle de redistribuer les paquets dans l'ordre au vocodeur. Si le paquet qui doit être délivré n'est pas encore arrivé mais que les quatre suivants le sont, le paquet est marqué comme perdu, mais le système de masquage de pertes de paquets peut profiter des quatre paquets situés dans le futur.

Quels que soient les systèmes envisagés, pour traiter la trame à l'instant t, il est nécessaire de disposer d'un certain nombre de trames de parole dans

le futur. Nous avons exploité ce tampon de compensation de gigue pour fixer le nombre de paquets disponibles dans le futur à 4 quel que soit le nombre de paquets manquants.

Dans le schéma 6.1, cela signifie que la variable J est égale à 4.

La représentation acoustique des trames utilisée est celle présentée dans le chapitre précédent, à savoir :

- le pourcentage de voisement, $v_{\%}$,
- 10 coefficients cepstraux de prédiction linéaire, $LPCC$,
- la dérivée du pourcentage de voisement, $\Delta v_{\%}$,
- la dérivée des LPCC, $\Delta LPCC$.

Le pourcentage de voisement ($v_{\%}$) et les coefficients cepstraux de prédiction linéaire ($LPCC$) du vecteur de paramétrisation acoustique $\boldsymbol{\phi}_t$ décrivant la trame t sont calculés sur une fenêtre d'analyse de 30ms de signal issu des trames $t-2$, $t-1$ et t. En effet, l'expérience nous a montré que la stabilité des paramètres calculés, notamment ceux issus de la prédiction linéaire, était mieux assurée sur une fenêtre 30ms que sur une fenêtre de 10ms. De plus, l'algorithme d'estimation du pourcentage de voisement repose sur une densité spectrale de puissance. Or la résolution fréquentielle de cette dernière dépend de la taille de la fenêtre d'analyse. Elle n'est pas suffisante pour une estimation correcte du pourcentage de voisement si la fenêtre d'analyse n'est pas d'au moins 200 échantillons (25ms à une fréquence d'échantillonnage de 8kHz).

Une fois les paramètres $LPCC$ et $v_{\%}$ estimés pour la trame t, les paramètres des trames $t-2$ et $t-1$ sont utilisés pour déterminer le vecteur d'observation acoustique $\boldsymbol{\phi}_{t-1}$ à l'aide de la formule de dérivation centrée rappelée ci-dessous pour un paramètre x :

$$\Delta x_{t-1} = \frac{x_t - x_{t-2}}{2}$$

Lors d'une trame manquante à l'instant τ, les dérivées des paramètres pour le vecteur d'observation précédent cette trame, $\boldsymbol{\phi}_{\tau-1}$, sont obtenues à

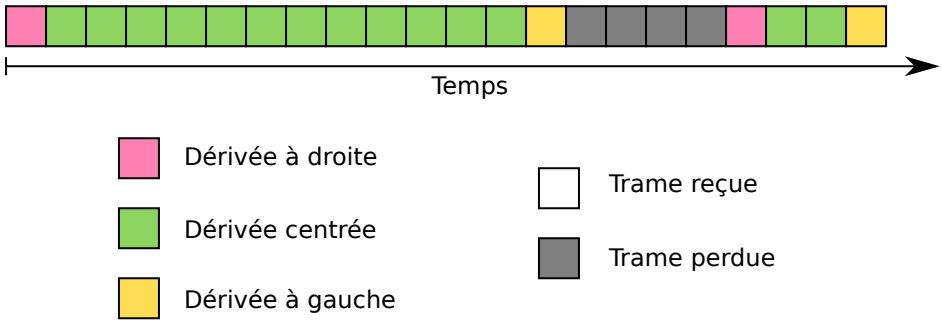


FIGURE 6.2 – Calcul des dérivées des paramètres lors de phénomènes de bord.

l'aide d'une formule de dérivation à droite des paramètres des trames $\tau - 1$, $\tau - 2$ et $\tau - 1$. Ceci est résumé sur la figure 6.2.

Pour éviter les discontinuités très gênantes pour l'oreille humaine lors de la phase de synthèse sonore, le signal est retardé de 5ms avant d'être délivré au système de reproduction afin de permettre le recouvrement[1] avec la première trame de synthèse lors de pertes de paquets et ainsi garantir une phase continue.

6.2 Architecture du système de masquage de pertes de paquets

La figure 6.3 présente l'architecture globale du système de masquage de pertes de paquets proposé. Deux cas de figure se posent :

la trame de signal est reçue : celle-ci est analysée les paramètres du modèles acoustique[2] (MMC) sont mis à jour, puis la trame est transmise au sous-système de restitution sonore.

la trame de signal est perdue : une estimation du vecteur de paramètres est réalisée. Le vecteur estimé est alors transmis à un module de syn-

[1] fondu enchainé

[2] la variable forward $\boldsymbol{\alpha}$ si la méthode de génération est `gmm`, les variables $\boldsymbol{\rho}$ et $\boldsymbol{\xi}$ si la méthode de génération est `viterbi`.

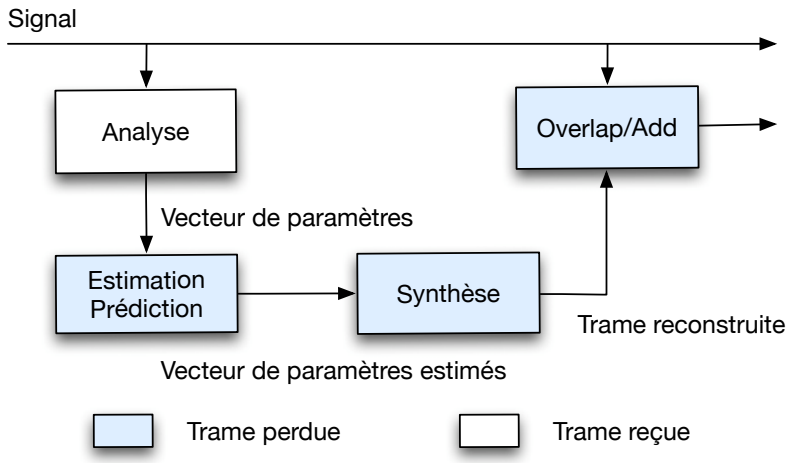


FIGURE 6.3 – Architecture du système global.

thèse de parole qui génère, à partir du vecteur estimé, une trame de signal qui est ensuite délivrée au sous-système de restitution sonore.

6.2.1 Mise à jour des paramètres du modèle acoustique (MMC)

Au fil du temps, la suite de trames est "décodée/estimée" par le modèle acoustique, à savoir un des modèles de Markov cachés proposés au chapitre précédent. Pour ce faire, à chaque réception d'une trame, $\boldsymbol{\phi}_t$ déterminé, les variables impliquées dans l'estimation sont mises à jour.

- Dans le cas des systèmes où l'estimation est faite à l'aide de mélanges de lois gaussiennes, la variable *forward* $\boldsymbol{\alpha}_t$ est mise à jour à l'aide de $\boldsymbol{\alpha}_{t-1}$, de la matrice de transition du modèle de Markov caché $\boldsymbol{A}$ et de la probabilité d'observer $\boldsymbol{\phi}_t$ conditionnellement à chaque loi d'observation du modèle, $b_1(\boldsymbol{\phi}_t) \cdots b_Q(\boldsymbol{\phi}_t)$.

- Dans le cas du système où l'estimation est réalisée à l'aide de l'algorithme de Viterbi modifié proposé au paragraphe 5.3.4, les vecteurs ρ_t et ξ_t sont mis à jour à l'aide des équations (5.29) et (5.30).

6.2.2 Estimation ou prédiction de la trame manquante

Lorsqu'une trame audio t vient à manquer du fait de la perte d'un paquet, une estimation du vecteur de paramétrisation $\hat{\boldsymbol{\phi}}_t$ est calculée puis transmise au système de synthèse de parole qui génère le signal manquant. Le calcul de $\hat{\boldsymbol{\phi}}_t$ peut être mené de trois manières différentes selon que l'estimation se fait à l'aide d'un mélange de gaussiennes, de la meilleure composante d'un mélange de lois gaussiennes ou à l'aide d'un décodage de Viterbi. Nous proposons de comparer ces approches de la manière suivante :

Mélange de lois gaussiennes. Cette approche noté `gmm` s'inspire de celle proposée par Christoffer Rødbro dans [25]. Néanmoins, notre système [52, 53] se distingue par les points suivants :

- l'utilisation d'un unique modèle de Markov caché. Nous avons, en partie grâce au pourcentage de voisement, supprimé la distinction voisée/non-voisée au niveau du modèle de Markov. Cette simplification permet de s'affranchir de toutes les transitions particulières entre les deux grands types de sons. Elle prend mieux en compte la nature continue du signal de parole.

- le vecteur de paramétrisation choisi. Celui-ci est guidé par le processus de synthèse. Le choix des coefficients de prédiction linéaire pour représenter la structure spectrale du signal rend le vecteur d'observation aisément utilisable pour une synthèse de parole de type source filtre.

De fait, $\hat{\boldsymbol{\phi}}_t$ s'exprime alors de la manière suivante :

$$\hat{\boldsymbol{\phi}}_t^{\mathrm{GMM}} = \sum_{i=1}^{Q} \mathrm{P}\left(q_t = i | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \boldsymbol{\mu}_i \tag{6.1}$$

Meilleure composante du mélange de lois gaussiennes. Cette variante notée `gmmbs` est une approximation de la précédente dans le sens où toutes les composantes du mélange de lois gaussiennes ne sont pas prises

en compte ; uniquement celle correspondant à l'état le plus probable $\tilde{q}_t$ au moment t de la trame perdue.

Ainsi,

$$\begin{aligned} \hat{\boldsymbol{\phi}}_t^{\text{GMMBS}} &= \boldsymbol{\mu}_{\tilde{q}_t} \\ \tilde{q}_t &= \underset{j}{\operatorname{argmax}}\, \mathrm{P}\left(q_t = j | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) \end{aligned} \tag{6.2}$$

Utilisation d'un décodage de Viterbi. L'une des contributions originales de ce travail est le recours à un décodage de Viterbi pour l'estimation du vecteur $\hat{\boldsymbol{\phi}}_t$. Cette méthode d'estimation est notée `Viterbi` et a été présentée dans le chapitre précédent.

6.2.3 Synthèse de parole

Le problème de la synthèse de parole est un large problème en soi qui pourrait faire l'objet de nombreuses études et développements. Dans le cadre de notre étude sur le masquage de pertes de paquets, nous ne pouvons pas prétendre à proposer un système optimal de synthèse de parole. Toutefois, afin de valider notre approche, il est nécessaire de choisir un système de synthèse et nous présentons dans ce paragraphe celui que nous avons implanté et qui nous est apparu le plus approprié vis-à-vis du système de masquage de pertes de paquets que nous proposons.

Le module de synthèse repose sur un modèle source-filtre : une fois le signal d'excitation généré, celui-ci est filtré à l'aide du filtre formantique estimé par l'une des trois méthodes exposées précédemment. Le schéma global de synthèse est présenté sur la figure 6.4.

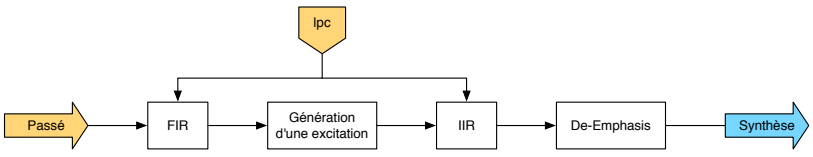


FIGURE 6.4 – Schéma de principe du module de synthèse de parole.

Afin de générer le signal résiduel des trames de paroles manquantes, on se base sur le signal de la dernière trame reçue. On utilise alors

- soit directement le signal d'excitation de la trame précédente périodisé par la valeur de la période fondamentale estimée sur les dernières trames. Cette méthode est décrite ci-après et est notée `sf` dans la suite de ce document.

- soit un algorithme de type ajout et recouvrement pitch synchrone de formes d'ondes (*Pitch Synchronous OverLap and Add* PSOLA) [54]. L'algorithme est également rappelé ci-après et est noté `psola`.

Périodisation à partir de la fréquence fondamentale

La génération de l'excitation à l'aide de la fréquence fondamentale est une méthode similaire à celle mise en œuvre dans le procédé de masquage de paquets de l'annexe 1 du codec G711 [2]. Un filtre autorégressif est estimé sur la dernière trame reçue. Cette dernière trame est alors filtrée à l'aide de ce filtre pour obtenir le signal résiduel. La dernière période de cette excitation est prolongée par périodisation sur la trame à générer. Ce procédé synthétise ainsi le signal d'excitation nécessaire à l'excitation du filtre formantique estimé par le modèle de Markov caché.

P.S.O.L.A.

La synthèse de parole par ajout et recouvrement synchrone de la période fondamentale (PSOLA) [54] est une méthode de synthèse de parole simple brevetée par le Centre National de Recherche en Télécommunications (CNET) en 1989 [55]. Elle est initialement destinée à être utilisée pour la synthèse à l'aide d'un corpus de diphones. Elle repose sur un fenêtrage centré sur le début de chaque réponse impulsionnelle correspondant à l'excitation du conduit vocal à l'aide d'une fenêtre de taille au moins deux fois celle de la période fondamentale. La superposition des différentes fenêtres permet de modifier

les paramètres prosodiques du signal de synthèse. Les fenêtres de recouvrement doivent être synchrones avec la fréquence fondamentale et centrées sur les maxima locaux d'énergie. Il est alors nécessaire d'effectuer au préalable un marquage de ces maxima locaux.

Marquage des maxima d'énergie du résiduel Afin de déterminer ces instants, nous avons utilisé l'algorithme de la boîte à outils de traitement du signal de l'université d'Edimbourg [56]. Les principales étapes sont les suivantes :

1. Filtrage RIF[3] passe bande entre 80 et $400Hz$

2. Filtre dérivateur

3. Marquage des passages par zéro en front descendant

La figure 6.5 donne le résultat d'un tel algorithme appliqué sur le résiduel d'un signal de parole.

Génération du signal d'excitation Une fois les maxima locaux d'énergie repérés, il est nécessaire, pour générer le signal d'excitation des trames manquantes, d'estimer la position des maxima locaux dans les trames manquantes puis d'apparier ces nouveaux maxima avec les anciens afin de pouvoir recréer le signal d'excitation.

Nous avons opté pour générer la position des maxima, d'utiliser la dernière valeur de la période fondamentale et d'apparier chaque nouvelle marque avec la dernière marque de la trame précédente.

Le signal ainsi généré sert d'excitation au filtre formantique estimé par le modèle de Markov caché.

[3]Réponse impulsionnelle finie

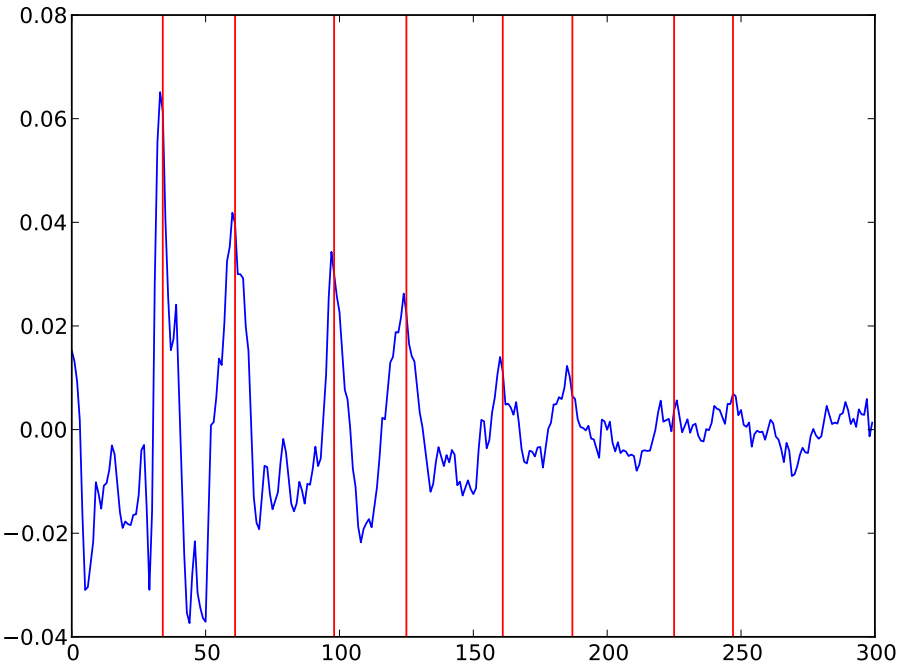


FIGURE 6.5 – Marquage des maxima d'énergie pitch-synchrone.

6.3 Évaluations

L'évaluation des solutions proposées au problème de masquage de pertes de paquets est faite en deux temps. Une première étape de comparaison utilisant l'erreur moyenne entre les observations estimées dans le cas de pertes de paquets et les observations sans pertes de paquets permet de valider la méthode d'estimation des observations manquantes.

La deuxième étape consiste à mesurer la dégradation produite par le masquage de pertes proposé et à la comparer aux dégradations des principaux algorithmes disponibles dans la littérature.

6.3.1 Protocole expérimental

Dans le but d'obtenir des résultats représentatifs et comparables, nous avons mis en place le protocole expérimental décrit dans le présent paragraphe. Il implique des règles de simulation des pertes et des mesures d'évaluation.

Position des trous et disponibilité du signal situé dans le "futur"

Afin de ne privilégier aucun son dans le choix des paquets perdus, nous avons adopté un schéma de pertes de paquets très structuré : pour un taux de trames perdues de $X\%$, nous considérons que X trames de signal consécutives sont perdues sur une seconde tel que présenté sur la figure 6.6. Bien qu'irréaliste, cette méthode de génération de pertes de paquets présente l'avantage de ne privilégier aucun son. De plus, pour un taux de $X\%$ de pertes de paquets, l'algorithme doit générer un signal équivalent à $X\%$ paquets consécutifs ce qui représente le cas le plus difficile par rapport à des motifs de pertes de paquets plus réalistes dans lesquelles les paquets perdus ne sont pas nécessairement consécutifs. Nous pouvons ainsi nous assurer du gain introduit par l'utilisation du modèle d'évolution temporel qu'est le modèle de Markov caché. Toutefois, l'annexe C présente un modèle réaliste de génération de pertes de paquets ayant fait l'objet d'une normalisation par l'ITU. Ce modèle normalisé n'a pas été utilisé dans le cadre des évaluations car nous avons constaté

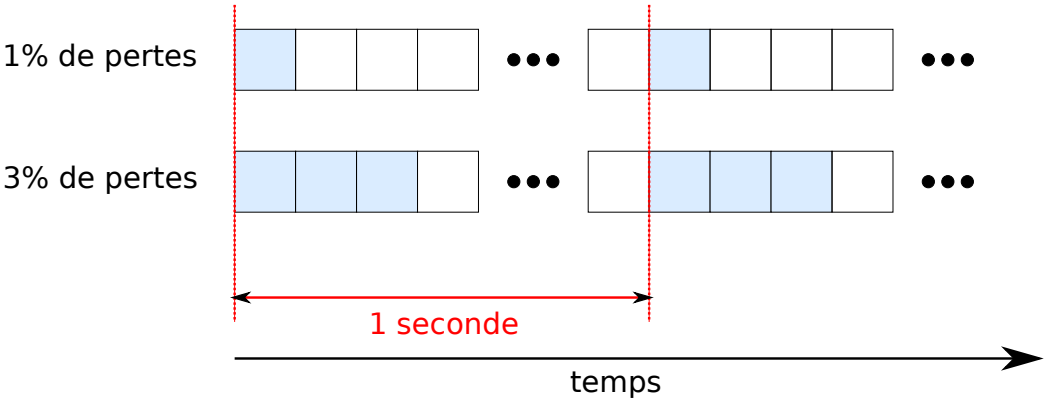


FIGURE 6.6 – Schéma de pertes de trames.
En bleu les paquets (trames de 10ms) considérés comme perdus, en blanc, les paquets considérés comme reçus. La figure présente deux exemples respectivement à des taux de 1% *et* 3% *de pertes de paquets.*

expérimentalement qu'il conduit à des pertes de peu de paquets consécutifs (un ou deux). De plus, la position des trames manquante est aléatoire et donc impossible à reproduire.

Comme nous l'avons introduit dans le chapitre 1, et repris dans ce chapitre, un certain nombre de paquets de parole peuvent être disponibles en avance dans le tampon de compensation de gigue. En règle générale, si le paquet qui doit être délivré n'est pas encore arrivé mais que les quatre suivants le sont, le paquet est marqué comme perdu, mais le système de masquage de pertes de paquets peut profiter des quatre paquets situés dans le futur. Toutes les expérimentations ont été menées dans ce cas de figure. Le nombre de paquets disponibles dans le futur a été fixé à 4 quel que soit le nombre de paquets manquants. Nous n'avons pas cherché à diminuer ce délai car il est déjà très faible, compte tenu des méthodes mises en œuvre ; l'augmenter devenait irréaliste pour l'application ciblée.

Distance entre observations et observations estimées

Afin d'évaluer la qualité de l'estimation proposée, nous avons mesuré l'erreur quadratique moyenne $\bar{e}$ entre les vecteurs estimés et les observations que l'on aurait dû avoir, à savoir les observations sans pertes de paquets $\boldsymbol{\phi}_\tau \cdots \boldsymbol{\phi}_{\tau+L-1}$, et les estimations $\hat{\boldsymbol{\phi}}_\tau \cdots \hat{\boldsymbol{\phi}}_{\tau+L-1}$ que l'on a estimées. Ainsi, $\bar{e}$ est défini de la manière suivante :

$$\bar{e} = \frac{1}{L} \sum_{t=\tau}^{\tau+L-1} \left\| \boldsymbol{\phi}_t - \hat{\boldsymbol{\phi}}_t \right\|^2 \tag{6.3}$$

Le choix de la distance euclidienne se justifie. En effet, d'après L. Rabiner [24], la distance euclidienne dans le domaine cepstral est équivalente à une distance spectrale. Or, $\boldsymbol{\phi}_t$ est composé des 10 premiers coefficients cepstraux issus de la prédiction linéaire, de leurs dérivées, du pourcentage de voisement et de sa dérivée.

Perceptual Evaluation of Speech Quality

Le calcul de l'erreur quadratique moyenne ne donne aucune information sur la qualité subjective du signal généré. C'est pourquoi, nous nous sommes également assurés de la qualité audio des algorithmes proposés au travers d'une mesure standard de la dégradation entre le signal sans perte et celui avec pertes.

Cette dégradation entre le signal sans pertes de paquets et la sortie du masquage de pertes est évaluée à l'aide d'un algorithme normalisé : le PESQ (*Perceptual Evaluation of Speech Quality*) [57] est un critère objectif d'évaluation de la qualité subjective de la parole. Il met en œuvre à la fois un modèle acoustique et un modèle psycho-acoustique afin de prédire la qualité sonore (et le confort) perçue par l'utilisateur d'un système de téléphonie. De ce fait, il tient compte de critères subjectifs à travers son modèle psycho-acoustique. Le PESQ produit pour chaque couple (signal de référence, signal dégradé) une valeur entre -0.5 et 4.5 prédisant la valeur du MOS[4] (Mean

[4]Mesure subjective de la dégradation introduite par un codec de parole. Sa valeur entre 1 et 5 est déterminée lors de tests subjectifs.

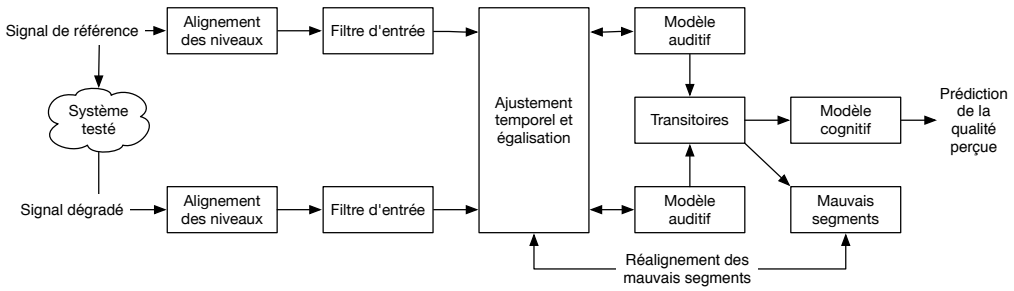


FIGURE 6.7 – Mise en œuvre du PESQ.
Inspirée de [59].

Opinion Score) [58].

La figure 6.7 montre les principales étapes de cet algorithme.

Son caractère normatif en fait une référence pour la comparaison des algorithmes de traitement de la parole téléphonique. A titre indicatif, les valeurs de PESQ des principaux codecs de parole sont donnés dans la table 6.1. Les dégradations acoustiques entre un signal sans pertes de trames et un signal produit par les systèmes proposés seront comparées à celles produites par trois systèmes présents dans l'état de l'art, basés respectivement sur :

- l'insertion de silences : chaque trame de signal perdu est remplacée par une trame de silence.
- le masquage comme indiqué dans l'annexe 1 de la norme du codeur G711 [2] qui est résumée dans le paragraphe 2.1.1.
- l'algorithme proposé par Gunduzhan et al.[18] décrit succintement dans le paragraphe 2.2.1.

6.3.2 Résultats

Nous présentons dans ce paragraphe les résultats expérimentaux de l'ensemble des expérimentations réalisées sur le masquage de pertes de paquets selon les deux aspects exposés dans le paragraphe 6.3.1.

Codecs		PESQ
G711	μ-Law	4.51
	A-Law	4.47
G726	16 kbits/s	3.06
	24 kbits/s	3.80
	32 kbits/s	4.22
	40 kbits/s	4.39
G728		4.15
G729		3.87
G729A		3.79
G723.1	5.3 kbits/s	3.54
	6.3 kbits/s	3.69

TABLE 6.1 – Valeurs de PESQ des principaux codecs de parole [60]. *Signaux de référence ITU en français. Ces valeurs sont directement extraites du document [60]. La valeur pour le codec G711 - μ-Law est bien* 4.51 *(>* 4.5*).*

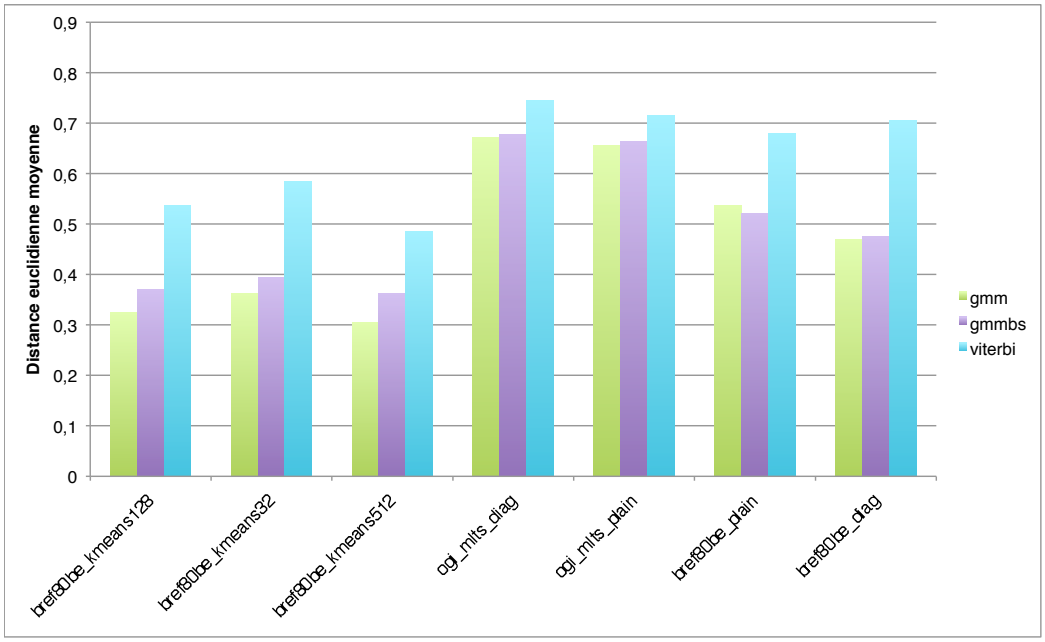


FIGURE 6.8 – Comparaison de la distance moyenne entre les observations estimées en cas de perte et les observations sans perte de trame.
Le taux de pertes de trame est de 1%.

Erreur d'estimation - indépendante du système de synthèse

Comme annoncé précédement, la qualité de l'estimateur de la trame acoustique est évaluée en comparant la distance euclidienne moyenne entre les observations estimées lors des pertes de paquets et les observations que l'on aurait eues sans perte de paquets.

Les trois figures 6.8, 6.9 et 6.10 montrent pour un taux de pertes de trames respectivement de 1%, 5% et 10%, cette distance euclidienne moyenne pour les sept modèles et les trois estimateurs proposés dans le paragraphe 6.2.2.

Il est clair que :

- Les deux approches GMM donnent des résultats supérieurs à l'approche Viterbi quel que soit le modèle. L'augmentation attendue de la distance moyenne lorsque seule la composante la plus probable du mélange de lois gaussiennes (approche `gmmbs`), par rapport à l'utilisation complète

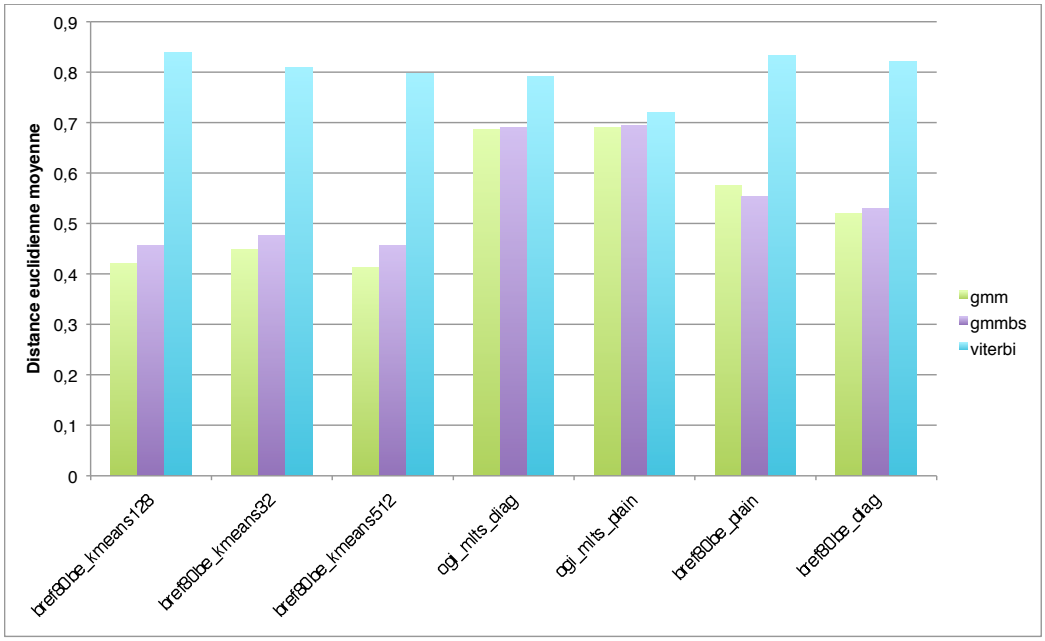


FIGURE 6.9 – Comparaison de la distance moyenne entre les observations estimées en cas de perte et les observations sans perte de trame.

Le taux de pertes de trame est de 5%.

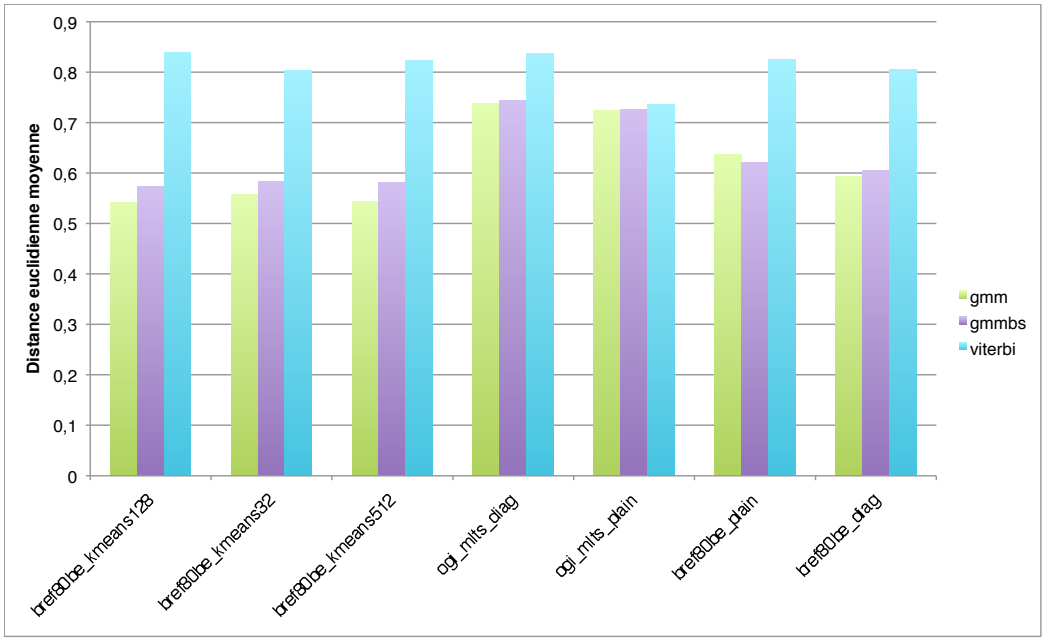


FIGURE 6.10 – Comparaison de la distance moyenne entre les observations estimées en cas de perte et les observations sans perte de trame.
Le taux de pertes de trame est de 10%.

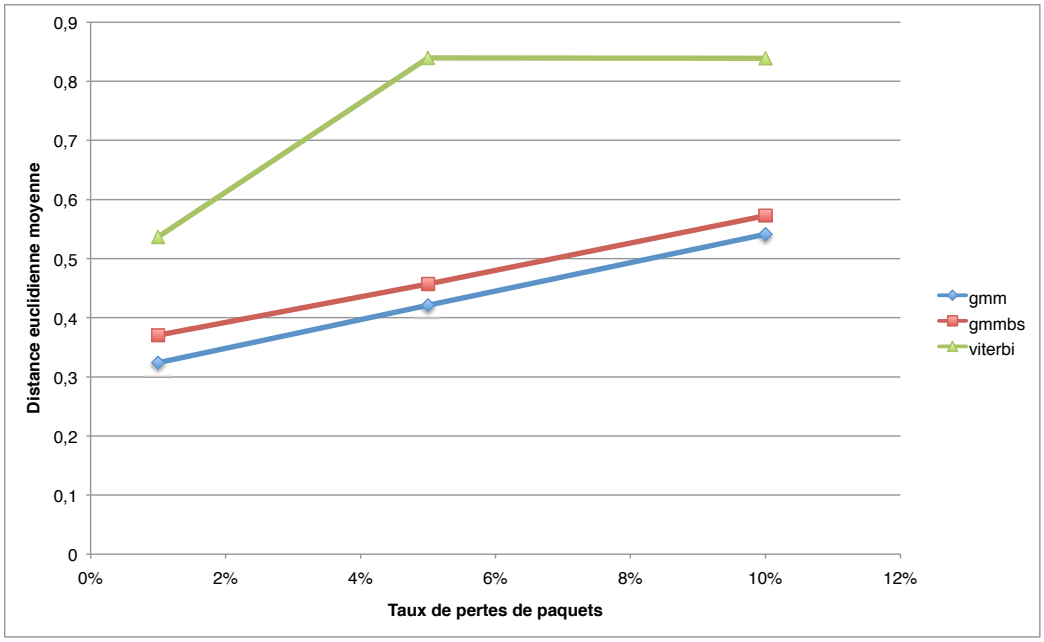


FIGURE 6.11 – Effet du taux de pertes de paquets sur l'erreur d'estimation. *Figure réalisée à l'aide du modèle* `bref80be_kmeans128`.

du mélange (approche `gmm`) est visible. Toutefois, cette augmentation reste faible et cela laisse supposer que seule la composante la plus probable est très nettement prédominante dans le mélange de lois gaussiennes ; les autres composantes n'ont pas de réel effet sur la qualité de l'estimation.

- Quel que soit le modèle de Markov servant de support à l'estimation, l'approche `viterbi` possède cependant des distances du même ordre de grandeur (aux alentours de 0.9).

- Les modèles de Markov cachés appris en mode non-supervisé donnent de meilleurs résultats que les modèles contraints, quelle que soit la méthode d'estimation. Le nombre d'états n'a que peu d'influence, et un nombre de 128 paraît un bon compromis quel que soit le taux de pertes.

La figure 6.11 nous montre que lors de pertes importantes de trames (5%

Taux de pertes	g711	gunduzhan	silence
1%	3,977	3,747	3,823
5%	3,376	3,298	2,763
10%	2,654	2,880	2,495

TABLE 6.2 – Performances des principaux algorithmes de masquage de pertes de paquets en terme de PESQ.

et 10%), la distance moyenne entre observations attendues et observations estimées semble progresser lentement.

Nous notons que l'ensemble des erreurs moyennes d'estimation lors de pertes de paquets sont du même ordre de grandeur que celles sans pertes de paquets qui ont été présentées dans le chapitre précédent. Par exemple, pour l'approche `viterbi`, la distance moyenne est de l'ordre de 0.9 que ce soit sans pertes de paquets ou avec pertes de paquets.

Une fois les performances de l'estimateur de vecteur acoustique vérifiées, une question reste en suspens : Qu'en est-il de l'amélioration de la qualité perçue du signal audio de parole ?

Afin d'évaluer cette qualité, le paragraphe suivant décrit les résultats en terme de PESQ en fonction des deux modules de synthèse proposés, des sept modèles acoustiques et des trois méthodes d'estimation envisagées.

Perceptual Evaluation of Speech Quality

Pour permettre les comparaisons et se situer par rapport aux algorithmes publiés dans la littérature, nous présentons dans le tableau 6.2 les performances en terme de PESQ des principaux algorithmes ayant fait l'objet d'une norme présentés dans le chapitre 2.

A la lecture du tableau 6.3 et de la figure 6.12, nous constatons que les résultats obtenus par nos différents systèmes sont très homogènes quels que soient le taux de pertes et la méthode utilisée. La comparaison avec l'état de l'art montre que nos systèmes sont dans tous les cas (sauf un) au moins aussi performants que les systèmes reconnus de l'état de l'art. Seul le système basé

Modèle	gmm		gmmbs		viterbi	
	psola	sf	psola	sf	psola	sf
bref80be_kmeans32	2,862	2,917	2,864	2,917	2,863	2,884
bref80be_kmeans128	2,867	2,916	2,865	2,916	2,863	2,881
bref80be_kmeans512	2,864	2,917	2,863	2,917	2,857	2,882
bref80be_diag	2,863	2,917	2,863	2,902	2,859	2,883
bref80be_plain	2,861	2,915	2,861	2,914	2,859	2,883
ogi_mlts_diag	2,865	2,902	2,864	2,902	2,860	2,919
ogi_mlts_plain	2,863	2,919	2,864	2,918	2,856	2,913

TABLE 6.3 – Valeurs de PESQ pour 10% de perte de paquets

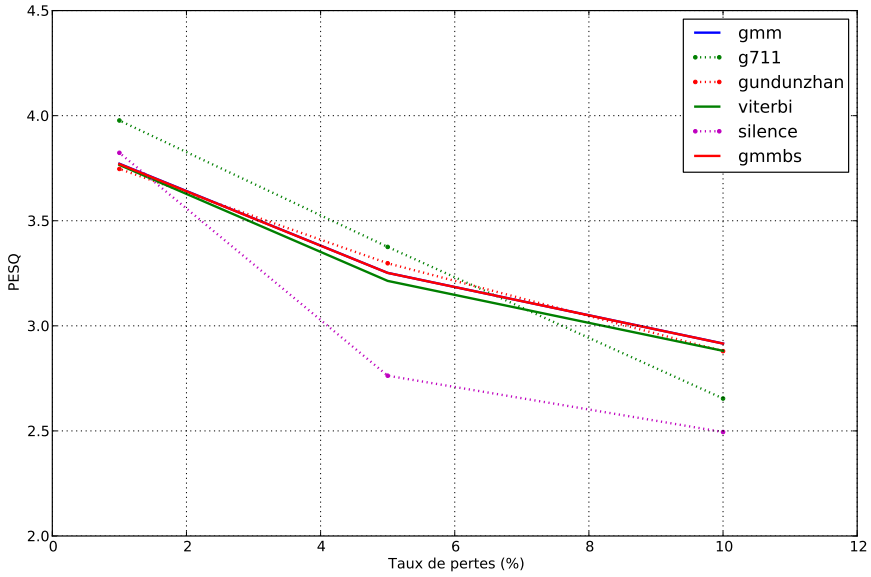


FIGURE 6.12 – Evolution du PESQ en fonction du taux de pertes de paquets. *Figure calculée pour le modèle* `bref80be_kmeans128` *et la méthode de synthèse source-filtre* `sf`. Remarque, les résultats GMM et GMMBS sont confondus.

G711 donne de meilleures performances dès lors que le taux de pertes reste limité à 1%.

Lors de longues pertes de trames, l'information apportée par la modélisation temporelle de l'évolution des paramètres acoustiques du signal apporte un gain. Pour un taux d'erreur de 10%, les valeurs de PESQ des algorithmes à base de modèles de Markov surpassent celles de l'état de l'art, l'information semble bien être retrouvée.

Nous remarquons également que les scores de PESQ sont indépendants de la méthode d'estimation. Ces valeurs de PESQ semblent dépendre quasiment uniquement de l'algorithme mis en œuvre pour la synthèse du signal d'excitation. L'origine de ce phénomène pourrait venir du fait que le PESQ compare deux signaux temporels. La valeur de PESQ calculée est très sensible à la modification de ces formes d'ondes. Or, les signaux temporels proviennent du module de synthèse et non directement du module d'estimation car, dans notre approche, seul le filtre formantique de synthèse est estimé à l'aide des méthodes proposées. Le PESQ semble mesurer principalement la qualité du module de synthèse et n'apporte que peu d'informations quant à celle de l'estimation.

L'annexe B comporte des résultats complémentaires, issus de nos évaluations qui recoupent les conclusions reportées dans ce paragraphe.

Cette série d'expérimentations nous a permis de valider la qualité globale du système. Elle assure que les méthodes proposées aboutissent à un résultat acceptable pour une tâche de téléphonie.

6.3.3 Discussion

Bien que l'approche `viterbi` semble moins performante que les approches `gmm` et `gmmbs` à la fois en terme de distance moyenne et de qualité subjective de la parole (PESQ), le coût calculatoire mis en œuvre est moindre : lors de l'estimation de la trame, seule la rétropropagation et la sélection de la moyenne correspondant à l'état le plus probable de la séquence sont nécessaires. Alors que, dans le cas des deux approches basées sur les mélanges de

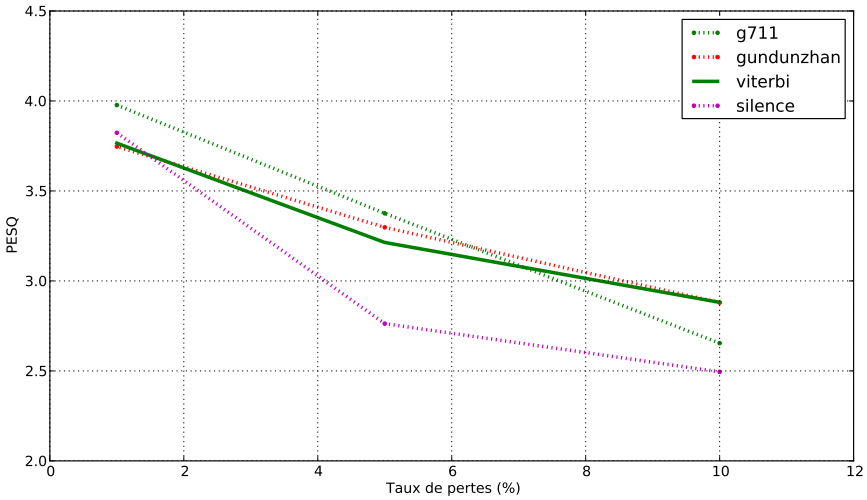


FIGURE 6.13 – Comparaison du système proposé avec l'état de l'art

lois gaussiennes (`gmm` et `gmmbs`), il est nécessaire d'effectuer une multiplication de la matrice de transition $\boldsymbol{A}$ avec elle-même (voir l'équation (2.7) dans le chapitre 2.3) ce qui augmente grandement le coût calculatoire.

Au vu des résultats présentés dans ce chapitre, nous préconisons l'utilisation du système utilisant l'approche `viterbi` pour l'estimation car elle implique des coûts calculatoires moins importants que les approches `gmm` ou `gmmbs`. L'utilisation du modèle `bref80be_kmeans128` est un bon compromis entre nombre d'états, donc coût calculatoire, et performances d'estimation. La méthode de synthèse `sf` est à la fois moins complexe et donne de meilleures performances que `psola` dans l'application visée.

La figure 6.13 compare les valeurs de PESQ du sytème proposé `viterbi/bref80be_kmeans128/sf` à celles des systèmes de l'état de l'art. On remarque que les performances de ce système sont comparables à celles du système proposé par Gunduzhan et al. Cette combinaison estimateur de Viterbi, modèle non-supervisé des 128 états et synthèse `sf` surpasse même l'algorithme du G711 lors de forts taux de pertes de paquets (10%).

Chapitre 7

Conclusions et perspectives

7.1 Conclusions

Dans le cadre de ce travail, nous proposons une approche originale pour résoudre le problème du masquage de pertes de paquets en voix sur IP au travers de plusieurs composants qui ont été présentés au fil de ce document. Agissant directement à partir du signal audio, en utilisant les propriétés acoustiques du signal de parole, le système que nous avons décrit dans ce document réalise l'opération complète d'estimation de paquets perdus. Ce système est par conséquent indépendant du codeur de parole utilisé pour la communication. L'utilisation des informations qui, grâce au tampon de compensation de gigue, sont situées dans le "futur" des observations manquantes, permet d'affiner le signal de parole restitué.

Pour construire ce système complet, nous avons étudié plusieurs composants présentés au fil de ce document :

- un estimateur original du degré de voisement d'un signal de parole : le pourcentage de voisement. De coût calculatoire faible, nous avons montré que cet estimateur était pertinent pour une tâche de segmentation voisée/non-voisée. Son utilisation comme paramètre acoustique complémentaire aux paramètres classiquement utilisés en reconnaissance de parole (MFCC, LPCC), permet une légère amélioration (0.5%) du taux

de reconnaissance phonétique d'un décodeur acoustico-phonétique. L'introduction du pourcentage de voisement dans le vecteur de représentation acoustique permet de s'affranchir de la distinction voisée/non-voisée.

- Un modèle acoustico-phonétique de type modèle de Markov caché. L'acoustique de la trame de signal de parole est représentée à l'aide d'un vecteur comportant à la fois le pourcentage de voisement et une représentation perceptuelle du spectre, les coefficients cepstraux issus de la prédiction linéaire (LPCC). Nous avons proposé de modéliser l'évolution stochastique de ce vecteur acoustique à l'aide d'un unique modèle de Markov caché ; cela permet, en cas de longues pertes de paquets, de mieux représenter l'évolution temporelle de cette suite.

  Nous avons utilisé, pour la construction du modèle de Markov caché modélisant l'évolution de la suite des vecteurs acoustique, deux approches. La première, dite *supervisée* permet d'utiliser les connaissances *a priori* sur la linguistique. La nécessité d'une expertise (annotations phonétiques) sur un vaste volume de signaux de parole rend cette approche difficile dans le cadre de la téléphonie où la variabilité de la langue est extrême. C'est pourquoi, dans un deuxième temps, nous avons proposé une approche *non-supervisée* à ce problème d'estimation du modèle. Cette méthode *non-supervisée* ne nécessite que la définition du nombre d'états.

- L'estimation des trames perdues. Une fois l'évolution temporelle du vecteur de paramétrisation acoustique représentée par un modèle de Markov caché, nous avons abordé le problème d'estimation des trames de signal perdues. Pour cela, nous avons introduit la notion de décodage acoustique d'un modèle de Markov caché lorsque certaines observations sont manquantes. Cela nous a amené a proposer une version modifiée de l'algorithme de Viterbi : un chemin optimal est trouvé conjointement aux observations manquantes.

7.2 Perspectives

Le présent document présente un instantané des travaux que nous avons réalisés dans le domaine du masquage de pertes de paquets. De nombreuses pistes d'améliorations et de perfectionnements peuvent être envisagées.

Modélisation Nous pensons par exemple à l'utilisation de deux modèles de Markov cachés : le premier serait dedié au suivi acoustique du signal reçu et le second, corrélé au premier, génèrerait les éléments nécessaires à la reconstruction. La génération serait plus adaptée à la synthèse de la parole, nous pourrions nous attendre à une amélioration du confort d'écoute. Cette approche est déjà mise en œuvre pour le suivi articulatoire [61].

L'utilisation de modèles de trajectoires couplés avec des modèles de Markov cachés comme ce qui est fait en matière de synthèse de parole [62] peut conduire à un meilleur suivi de l'évolution acoustique de la parole et ainsi, à une meilleure reconstruction des trames manquantes.

Pourcentage de voisement L'estimation de la puissance du bruit dans l'algorithme du pourcentage de voisement peut être améliorée : les travaux de Mathieu Durnerin présenté dans sa thèse [38] vont plus loin que la simple utilisation d'un filtre médian pour estimer la ligne de fond de spectre.

L'utilisation du pourcentage de voisement comme entrée d'un synthétiseur de parole est une piste qui n'a pas encore été explorée. Nous pouvons imaginer que le pourcentage de voisement serve à pondérer la partie harmonique de la partie bruit lors d'une synthèse harmonique plus bruit.

Synthèse de parole Comme nous l'avons montré dans le chapitre 6, la qualité subjective du signal audio produit par le système ne dépend quasiment que du module de synthèse utilisé. Nous pensons qu'améliorer les méthodes mises en œuvre dans cette partie du système peut conduire à un grand bond dans le confort engendré pour les utilisateurs.

Système de masquage de pertes de paquets en voix sur IP Bien que l'étude ait porté sur la téléphonie en réseau IP, les travaux présentés peuvent être étendus à d'autres domaines nécessitant l'utilisation de modèle de Markov caché en environnement contraignant. Nous pensons en particulier à la reconnaissance de la parole lorsque la transmission du signal entre le locuteur et le système de reconnaissance n'est pas fiable, notamment dans le cadre d'un service de réservation par téléphone.

Annexe A

Détermination de la probabilité d'être dans l'état i connaissant les observations reçues

Nous présentons dans cette annexe les développements des calculs du paragraphe 2.3 permettant de passer de l'équation (2.6) à l'équation (2.7). Le point de départ est la probabilité d'être dans un état i au moment $\tau + k$ avec $0 \leq k < L$ connaissant les observations $\boldsymbol{\phi}_1 \cdots \boldsymbol{\phi}_{\tau-1}$ et $\boldsymbol{\phi}_{\tau+L} \cdots \boldsymbol{\phi}_{\tau+L+J-1}$ où, nous le rappelons, L est le nombre d'observations manquantes et J est le nombre d'observations disponibles après la perte de paquets.

Autrement dit, nous cherchons à déterminer la quantité suivante.

$$\mathrm{P}\left(q_{\tau+k} = i | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+J+L-1}\right) \qquad 0 \leq k < L \tag{A.1}$$

Nous rappelons que $\boldsymbol{A} = (a_{ij})_{(i,j)\in[\![1;Q]\!]^2}$ est la matrice de transition du modèle de Markov caché :

$$a_{i,j} = \mathrm{P}\left(q_t = j | q_{t-1} = i\right)$$

A partir de l'équation (A.1) et en utilisant la loi de Bayes, nous obtenons :

$$\mathrm{P}\left(q_{\tau+k} = i | \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right) = \frac{\mathrm{P}\left(q_{\tau+k} = i, \boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right)}{\mathrm{P}\left(\boldsymbol{\phi}_1^{\tau-1}, \boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right)} \tag{A.2}$$

Or

$$\mathrm{P}\left(q_{\tau+k}=i,\boldsymbol{\phi}_1^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}\right)= \\ \mathrm{P}\left(q_{\tau+k}=i,\boldsymbol{\phi}_1^{\tau-1}\right)\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+k}=i,\boldsymbol{\phi}_1^{\tau-1}\right) \quad (A.3)$$

En exploitant la propriété de Markov, nous pouvons développer le premier terme du membre de droite de l'équation (A.3) pour $k=0$:

$$\mathrm{P}\left(q_\tau=i,\boldsymbol{\phi}_1^{\tau-1}\right)=\sum_{j=1}^{Q}\mathrm{P}\left(q_\tau=i,q_{\tau-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right) \quad (A.4)$$

$$=\sum_{j=1}^{Q}\underbrace{\mathrm{P}\left(q_\tau=i|q_{\tau-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right)}_{a_{j,i}}\underbrace{\mathrm{P}\left(q_{\tau-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right)}_{\alpha_{\tau-1}(j)} \quad (A.5)$$

avec $\alpha_{\tau-1}(j)=\mathrm{P}\left(q_{\tau-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right)$ la variable *forward* associée au modèle de Markov caché.

Ainsi, si nous notons $\boldsymbol{\aleph}_k$ le vecteur $\left[\mathrm{P}\left(q_{\tau+k}=i,\boldsymbol{\phi}_1^{\tau-1}\right)\right]_{i\in[\![1,Q]\!]}^{\dagger}$, nous pouvons réécrire (A.5) sous forme vectorielle :

$$\boldsymbol{\aleph}_0=\boldsymbol{A}\boldsymbol{\alpha}_{\tau-1} \quad (A.6)$$

avec $\boldsymbol{\alpha}_{\tau-1}=[\alpha_{\tau-1}(1),\cdots,\alpha_{\tau-1}(Q)]^{\dagger}$.

Puis, pour $k=1$,

$$\mathrm{P}\left(q_{\tau+1}=i,\boldsymbol{\phi}_1^{\tau-1}\right)=\sum_{j=1}^{Q}\mathrm{P}\left(q_{\tau+1}=i,q_\tau=j,\boldsymbol{\phi}_1^{\tau-1}\right) \\ =\sum_{j=1}^{Q}\underbrace{\mathrm{P}\left(q_{\tau+1}=i|q_\tau=j,\boldsymbol{\phi}_1^{\tau-1}\right)}_{a_{j,i}}\mathrm{P}\left(q_\tau=j,\boldsymbol{\phi}_1^{\tau-1}\right) \quad (A.7)$$

En utilisant l'écriture vectorielle proposée ci-dessus, nous avons :

$$\begin{aligned}\boldsymbol{\aleph}_1=&\boldsymbol{A}\boldsymbol{\aleph}_0\\ =&\boldsymbol{A}^2\boldsymbol{\alpha}_{\tau-1}\end{aligned} \quad (A.8)$$

Enfin, pour tout k tel que $0 \leq k < L$, nous pouvons écrire :

$$\begin{aligned}
\mathrm{P}\left(q_{\tau+k}=i,\boldsymbol{\phi}_1^{\tau-1}\right) &= \sum_{j=1}^{Q} \mathrm{P}\left(q_{\tau+k}=i,q_{\tau+k-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right) \\
&= \sum_{j=1}^{Q} \underbrace{\mathrm{P}\left(q_{\tau+k}=i|q_{\tau+k-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right)}_{a_{j,i}} \\
&\qquad \underbrace{\mathrm{P}\left(q_{\tau+k-1}=j,\boldsymbol{\phi}_1^{\tau-1}\right)}_{\boldsymbol{\aleph}_{k-1}(j)} \qquad (A.9)
\end{aligned}$$

En réécrivant l'équation (A.9) sous forme vectorielle, on en déduit par récurrence que pour tout $0 \leq k < L$:

$$\begin{aligned}
\boldsymbol{\aleph}_k &= \boldsymbol{A}\boldsymbol{\aleph}_{k-1} \\
&= \boldsymbol{A}^{k+1}\boldsymbol{\alpha}_{\tau-1} \qquad (A.10)
\end{aligned}$$

Le deuxième terme du membre de droite de l'équation (A.3) est développé ci dessous. Tout d'abord, remarquons que, du fait de la propriété de Markov, pour $k = L-1$ nous avons :

$$\begin{aligned}
\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-1}=i,\boldsymbol{\phi}_1^{\tau-1}\right) &= \mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-1}=i\right) \qquad (A.11) \\
&= \beta_{\tau+L-1}(i) \qquad (A.12)
\end{aligned}$$

où pour tout t, $\beta_{\tau+L-1}(i) = \mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-1}=i\right)$

Puis pour $k = L-2$,

$$\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-2}=i\right) = \sum_{j=1}^{Q} \mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1},q_{\tau+L-1}=j|q_{\tau+L-2}=i\right) \qquad (A.13)$$

Or, $\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1},q_{\tau+L-1}=j|q_{\tau+L-2}=i\right)$ peut être développé en :

$$\begin{aligned}
&\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1},q_{\tau+L-1}=j|q_{\tau+L-2}=i\right) = \\
&\quad \mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-1}=j,q_{\tau+L-2}=i\right) \underbrace{\mathrm{P}\left(q_{\tau+L-1}=j|q_{\tau+L-2}=i\right)}_{a_{ij}} \qquad (A.14)
\end{aligned}$$

En remarquant que :

$$\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-1}=j,q_{\tau+L-2}=i\right)=\beta_{\tau+L-1}(j) \tag{A.15}$$

nous pouvons écrire :

$$\begin{aligned}\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+L-2}=i\right)&=\sum_{j=1}^{Q}a_{ij}\beta_{\tau+L-1}(j)\\&=\left(\boldsymbol{A}^{\dagger}\boldsymbol{\beta}_{\tau+L-1}\right)_i\end{aligned} \tag{A.16}$$

Ce qui se généralise en utilisant le même principe de récurrence que précédemment en :

$$\begin{aligned}\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|\boldsymbol{\phi}_{1}^{\tau-1},q_{\tau+k}=i\right)&=\mathrm{P}\left(\boldsymbol{\phi}_{\tau+L}^{\tau+L+J-1}|q_{\tau+k}=i\right)\\&=\left(\left(\boldsymbol{A}^{\dagger}\right)^{L-k-1}\boldsymbol{\beta}_{\tau+L-1}\right)_i\end{aligned} \tag{A.17}$$

En combinant les équations(A.2), (A.3), (A.10) et (A.17) nous pouvons écrire l'équation (2.7) rappelée ci-dessous.

$k=0,\cdots,L-1$

$$\mathrm{P}\left(q_{\tau+k}=i|\boldsymbol{\phi}_{1}^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+J+L}\right)=\frac{\left(\boldsymbol{A}^{(k+1)}\boldsymbol{\alpha}_{\tau-1}\right)_i\left(\left(\boldsymbol{A}^{\dagger}\right)^{(L-k-1)}\boldsymbol{\beta}_{\tau+L}\right)_i}{\mathrm{P}(\boldsymbol{\phi}_{1}^{\tau-1},\boldsymbol{\phi}_{\tau+L}^{\tau+L+J})} \tag{A.18}$$

Annexe B

Résultats complémentaires

Cette annexe présente quelques résultats complémentaires à ceux présentés dans le paragraphe 6.3.2.

B.1 Erreur d'estimation

Les figures B.1, B.2 et B.3 présentent l'évolution de la distance moyenne en fonction du taux de pertes de paquets.

Nous remarquons que, plus le taux de pertes de paquets augmente, plus la distance moyenne entre les observations attendues et les observations estimées augmente quel que soit le modèle acoustique ou l'estimateur utilisé. Notons tout de même que les performances de l'estimateur `viterbi` pour un taux de perte de paquets de 10% (figure B.3) sont identiques à celles pour un taux de 5%. Ce résultat nous conforte sur les performances de l'estimateur de Viterbi pour d'importants taux de pertes.

B.2 PESQ

Les tableaux B.1 et B.2 complètent le tableau 6.3 présenté dans le paragraphe 6.3.2.

A la lecture de ces deux tableaux, nous remarquons que les scores de

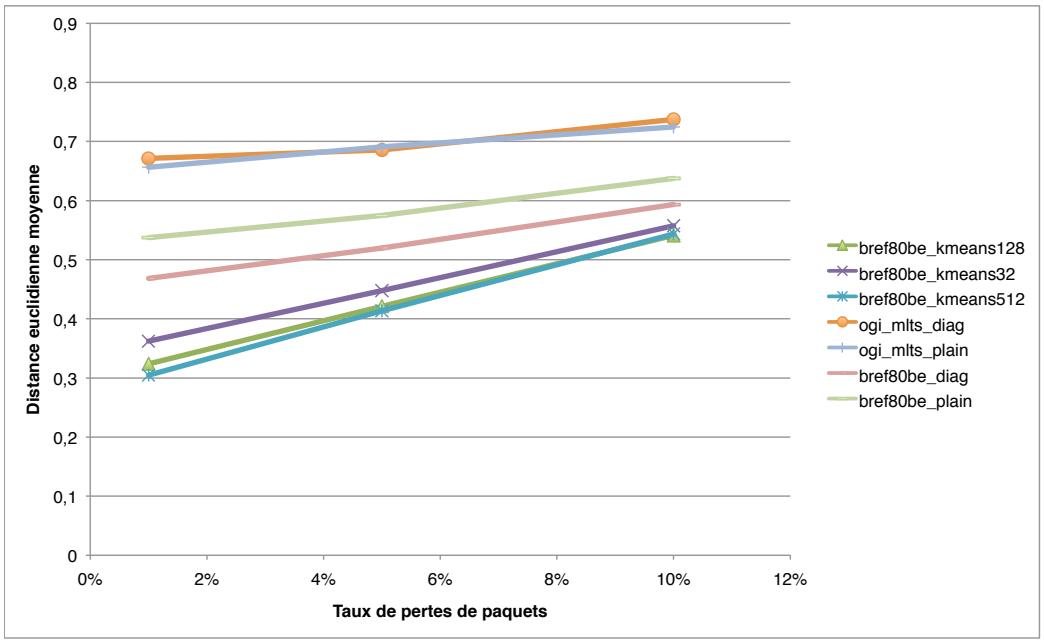


FIGURE B.1 – Comparaison de la distance moyenne en fonction du modèle et du taux de perte de paquets lorsque l'estimateur est `gmm`.

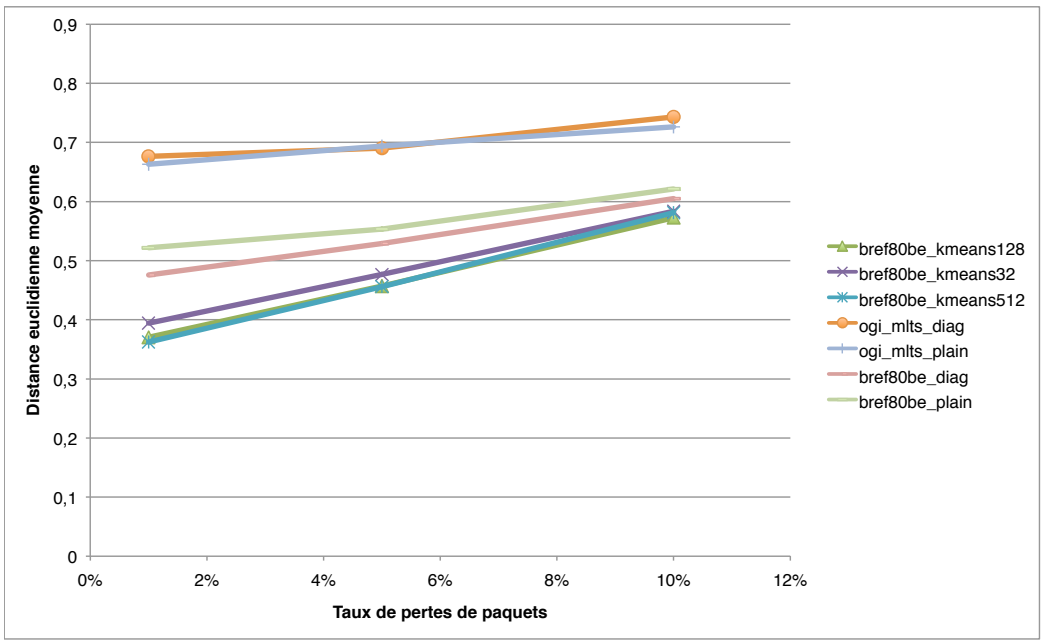


FIGURE B.2 – Comparaison de la distance moyenne en fonction du modèle et du taux de perte de paquets lorsque l'estimateur est `gmmbs`.

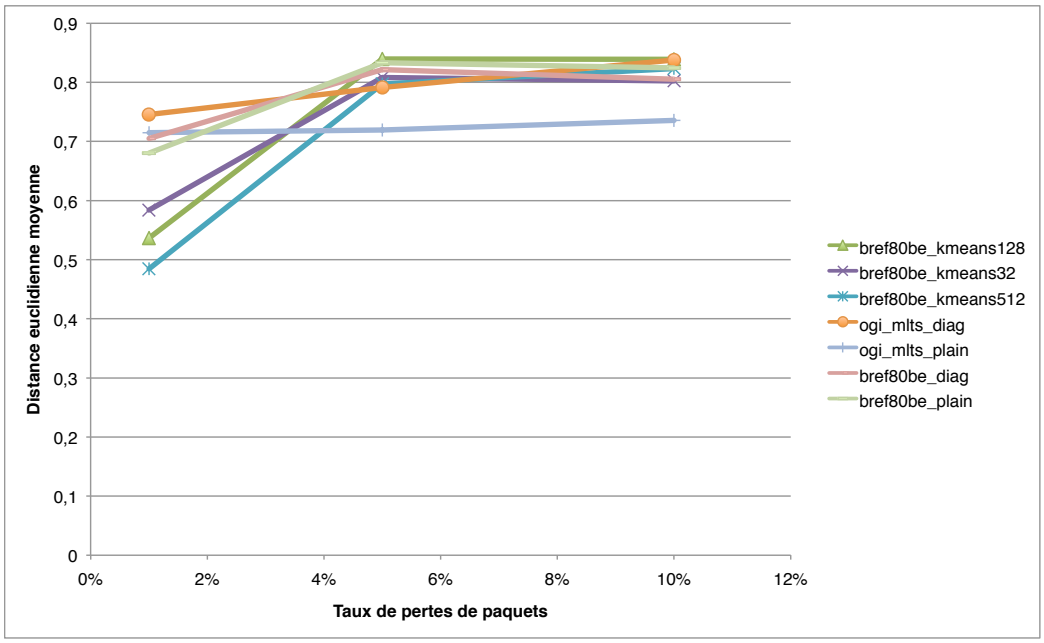


FIGURE B.3 – Comparaison de la distance moyenne en fonction du modèle et du taux de perte de paquets lorsque l'estimateur est `viterbi`.

Modèle	`gmm`		`gmmbs`		`viterbi`	
	`psola`	`sf`	`psola`	`sf`	`psola`	`sf`
bref80be_kmeans32	3,818	3,772	3,818	3,771	3,817	3,759
bref80be_kmeans128	3,822	3,771	3,822	3,770	3,822	3,766
bref80be_kmeans512	3,819	3,771	3,817	3,770	3,819	3,771
bref80be_diag	3,822	3,773	3,821	3,773	3,816	3,755
bref80be_plain	3,820	3,770	3,820	3,769	3,817	3,758
ogi_mlts_diag	3,819	3,777	3,821	3,775	3,819	3,749
ogi_mlts_plain	3,818	3,774	3,817	3,774	3,817	3,773

TABLE B.1 – Valeurs de PESQ pour 1% de perte de paquets.

Modèle	gmm		gmmbs		viterbi	
	psola	sf	psola	sf	psola	sf
bref80be_kmeans32	3,229	3,252	3,231	3,252	3,230	3,199
bref80be_kmeans128	3,231	3,252	3,231	3,252	3,231	3,214
bref80be_kmeans512	3,230	3,252	3,231	3,252	3,228	3,219
bref80be_diag	3,231	3,253	3,230	3,253	3,228	3,215
bref80be_plain	3,229	3,249	3,229	3,248	3,228	3,216
ogi_mlts_diag	3,228	3,258	3,229	3,257	3,231	3,239
ogi_mlts_plain	3,230	3,256	3,231	3,256	3,230	3,255

TABLE B.2 – Valeurs de PESQ pour 5% de perte de paquets.

PESQ sont peu dépendants de la méthode d'estimation. Nous notons également que pour de longues pertes de paquets (taux d'erreurs de 10%), l'approche `viterbi` semble être aussi robuste que les approches `gmm` et `gmmbs`.

Annexe C

Modèle de pertes de paquets de Gilbert-Elliott

Les systèmes de transmission de données par paquets sont soumis aux phénomènes de pertes. Comme nous l'avons vu précédemment, ces pertes sont dues soit à une congestion à un noeud du réseau, soit, dans le cadre de la téléphonie sur IP à un trop long retard d'un paquet qui est alors détruit, un lien réseau coupé. Dans les réseaux IP, la perte de paquets est par nature en rafale. On distingue la perte en rafale de la perte de paquets consécutifs dans le sens où la perte en rafale est une période de temps où une grande majorité de paquets sont perdus alors que lors d'une perte de paquets consécutifs, tous les paquets sont perdus.

Les pertes de paquets sont modélisées à l'aide d'une chaîne de Markov à deux états selon l'approche proposée par Gilbert et Elliott [63]. La figure C.1 présente ce modèle.

À ce modèle, on associe quatre paramètres :

p est la probabilité de passer d'un état de faibles pertes à un état de pertes en rafales

q est la probabilité de passer d'un état de pertes élevées "Rafale" à un état dans lequel les paquets sont faiblement perdus.

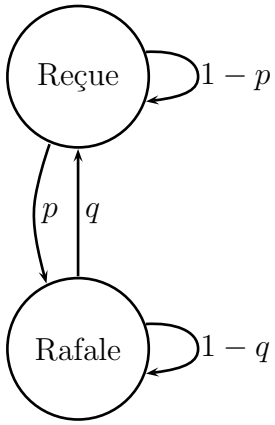


FIGURE C.1 – Modèle de Gilbert-Elliot

P_{good} est la probabilité de perdre un paquet lorsque l'on est dans l'état "Reçue". Celle-ci est très faible (≈ 0).

P_{bad} est la probabilité de perdre un paquet lorsque l'on est dans l'état "Rafale". Celle-ci est plutôt élevée (≈ 0.5).

Le taux de paquets perdus moyen, *Frame Erasure Rate* (FER), est alors donné par la relation suivante :

$$FER = \frac{p}{1-\kappa} P_{\text{bad}} + \frac{q}{1-\kappa} P_{\text{good}} \tag{C.1}$$

où

$$\kappa = 1 - (p + q)$$

κ est alors une mesure du caractère groupé de la perte des paquets : si κ est proche de 1 les paquets sont perdus de manière corrélée alors que si κ est proche de 0, les paquets sont plutôt perdus de manière éparse.

Les valeurs recommandées par l'ITU sont les suivantes :

$$\begin{aligned} P_{\text{good}} &= 0.0 \\ P_{\text{bad}} &= 0.5 \end{aligned}$$

Avec ces valeurs, on peut déterminer les valeurs de p et de q à l'aide du taux d'erreurs voulu FER et du paramètre de corrélation κ :

$$\begin{aligned} p &= 2(1-\kappa)FER \\ q &= (1-\kappa)(1-2FER) \end{aligned}$$

Le lecteur intéressé pourra trouver dans l'article de A.N. Gilbert [63] plus d'information sur ce modèle. Des détails d'implémentations sont disponibles dans la recommandation G.191 [64] de l'ITU.

Table des figures

Liste des tableaux

Bibliographie

[1] ITU-T study group 12, "One-way transmission time," tech. rep., International Telecommunication Union, May 2003.

[2] ITU-T study group 16, "Pulse code modulation (PCM) of voice frequencies," tech. rep., International Telecommunication Union, Nov 1988.

[3] ITU-T study group 16, "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (cs-acelp)," tech. rep., International Telecommunication Union, 2007.

[4] ITU-T study group 16, "Wideband coding of speech at around 16 kbit/s using adaptive multi-rate wideband (amr-wb)," tech. rep., International Telecommunication Union, Jul 2003.

[5] S. Andersen, S. Andersen, W. Kleijn, R. Hagen, J. Linden, M. Murthi, and J. Skoglund, "iLBC - a linear predictive coder with robustness to packet losses," in *Speech Coding, 2002, IEEE Workshop Proceedings.*, pp. 23–25, 2002.

[6] S. Andersen, A. Duric, Telio, H. Astrom, R. Hagen, W. Kleijn, and J. Linden, "Internet low bit rate codec (ilbc)," *Network Working Group Request for Comments*, vol. 3951, p. 1, Dec 2004.

[7] J. C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive FEC-based error control for Internet telephony," in *INFOCOM '99. Eighteenth Annual*

Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, (New York, NY , USA), pp. 1453–1460.

[8] J. Lindblom, J. Lindblom, and P. Hedelin, "Error protection and packet loss concealment based on a signal matched sinusoidal vocoder," vol. 1, pp. I–100–I–103 vol.1, 2003.

[9] M. Chen and M. N. Murthi, "Optimized unequal error protection for voice over IP," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, vol. 5, pp. 865–8, 2004.

[10] V. Koen, J. Sozeren Skak, and S. Karsten Vandborg, "Silk speech codec," draft, The Internet Engineering Task Force (IETF), Mar 2010.

[11] ITU-T study group 16, "7 khz audio-coding within 64 kbit/s," tech. rep., International Telecommunication Union, 1988.

[12] ITU-T study group 16, "Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s," tech. rep., International Telecommunication Union, 2006.

[13] ITU-T study group 16, "Coding of speech at 16 kbit/s using low-delay code excited linear prediction," tech. rep., International Telecommunication Union, 1992.

[14] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 111, p. 1917, Apr 2002.

[15] S. Adrian and N. Mihai, "Generating a frame of audio data," 1989.

[16] K. Kondo and K. Nakagawa, "A speech packet loss concealment method using linear prediction," *IEICE Transactions on Information and Systems*, vol. E89-D, no. (2), pp. 806–813, 2006.

[17] N. Levinson, "The Wiener RMS error criterion in filter design and prediction," *Journal of Mathematics and Physics*, vol. 25, no. 4, pp. 261–278, 1947.

[18] E. Gunduzhan and K. Momtahan, "Linear prediction based packet loss concealment algorithm for pcm coded speech," *Speech and Audio Processing, IEEE Transactions on*, vol. 9, no. 8, pp. 778–785, 2001.

[19] E. Zavarehei and S. Vaseghi, "Interpolation of lost speech segments using LP-HNM model with codebook-mapping post-processing," in *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, pp. 14–18, 2007.

[20] E. Zavarehei and S. Vaseghi, "Interpolation of lost speech segments using lp-hnm model with codebook post-processing," *IEEE Transactions on Multimedia*, vol. 10, no. 3, pp. 493–502, 2008.

[21] J. Lindblom and P. Hedelin, "Packet loss concealment based on sinusoidal extrapolation," in *Acoustics, Speech, and Signal Processing, 2002. Proceedings. (ICASSP '02). IEEE International Conference on*, vol. 1, pp. I–173–I–176 vol.1, 2002.

[22] J. Lindblom and P. Hedelin, "Packet loss concealment based on sinusoidal modeling," in *Speech Coding, 2002, IEEE Workshop Proceedings.*, pp. 65–67, 2002.

[23] J. Lindblom and J. Lindblom, "A sinusoidal voice over packet coder tailored for the frame-erasure channel," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 787–798, 2005.

[24] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. Upper Saddle River, NJ, USA : Prentice-Hall, Inc., alan v.oppenheim, series editor ed., 1993.

[25] C. Rodbro, M. Murthi, S. Andersen, and S. Jensen, "Hidden markov model-based packet loss concealment for voice over ip," *Audio, Speech*

and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on], vol. 14, no. 5, pp. 1609–1623, 2006.

[26] M. Murthi, C. Rodbro, S. Andersen, and S. Jensen, "Packet Loss Concealment with Natural Variations using HMM," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, pp. I–I, 2006.

[27] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[28] F. Itakura, "Line spectrum representation of linear predictor coefficients of speech signals," vol. 57, p. 35, 1975.

[29] L. Lamel, J.-L. Gauvain, and M. Eskénazi, "BREF, a large vocabulary spoken corpus for French," in *Proc. Eurospeech*, pp. 505–508, 1991.

[30] O. Leblouch, *Décodage acoustico-phonétique et application à l'indexation audio automatique*. PhD thesis, Université de Toulouse 3 - Paul Sabatier, 2009.

[31] Y. K. Muthusamy, R. A. Cole, and B. T. Oshika, "The ogi multilanguage telephone speech corpus," tech. rep., Center of Spoken Language Understanding, Oregon Graduate Institute of Technology, Oct 1992.

[32] T. Lander and J. L. Hieronymus, "The cslu labeling guide," tech. rep., 1997.

[33] W. Hess, *Pitch Determination of Speech Signals*. Springer-Verlag, 1983.

[34] Y. D. Cho, H. K. Kim, M. Y. Kim, and S. R. Kim, "Pitch estimation using spectral covariance method for low-delay MBE vocoder," in *Speech Coding For Telecommunications Proceeding, 1997, 1997 IEEE Workshop on*, (Pocono Manor, PA , USA), pp. 21–22, 1997.

[35] Y. D. Cho, M. Y. Kim, and S. R. Kim, "A spectrally mixed excitation (SMX) vocoder with robust parameter determination," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 2, (Seattle, WA), pp. 601–604, 1998.

[36] W. Chou and L. Gu, "Robust singing detection in speech/music discriminator design," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 2, (Salt Lake City, UT), pp. 865–868, 2001.

[37] L. Koenig, C. Mailhes, R. André-Obrecht, and S. Fabre, "A continuous voicing parameter in the frequency domain," in *International Conference on Speech and Computer (SPECOM)*, Jun 2009.

[38] M. Durnerin, *Une stratégie pour l'interprétation en analyse spectrale, détection et caractérisation des composantes d'un spectre.* PhD thesis, INPG, 1999.

[39] R. Andre-Obrecht, "A new statistical approach for the automatic segmentation ofcontinuous speech signals," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 1, pp. 29–40, 1988.

[40] J.-L. Gauvain and L. F. Lamel, "Speaker-independent phone recognition using BREF," 1992.

[41] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*, vol. 3.4. Cambridge University Engineering Department, 2006.

[42] D. Jouvet, *Reconnaissance de mots connectés indépendamment du locuteur par des méthodes statistiques.* PhD thesis, ENST, 1988.

[43] L. Lamel and J. Gauvain, "Language identification using phone-based acoustic likelihoods," *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 1, pp. 293–296, 1994.

[44] M. A. Zissman, "Automatic language identification using Gaussian mixture and hidden Markov models," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, (Minneapolis, MN , USA), pp. 399–402, 1993.

[45] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 131–142, Aug 2002.

[46] H. Zen, K. Tokuda, and T. Kitamura, "A viterbi algorithm for a trajectory model derived from hmm with explicit relationship between static and dynamic features," vol. 1, pp. 837–40, 2004.

[47] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039 – 1064, 2009.

[48] J. Tubach and Calliope, *La Parole et son traitement automatique*. Masson, 1989.

[49] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, 1990.

[50] C. Corredor-Ardoy, L. Lamel, M. Adda-Decker, and J. L. Gauvain, "Multilingual phone recognition of spontaneous telephone speech," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, (Seattle, WA , USA), pp. 413–416.

[51] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd Edition*. Wiley, 2000.

[52] L. Koenig, R. André-Obrecht, C. Mailhes, and S. Fabre, "A new feature vector for hmm-based packet loss concealment," in *European Signal Processing Conference*, Aug 2009.

[53] L. Koenig, R. André-Obrecht, C. Mailhes, and S. Fabre, "Modèles de markov cachés appliqués au masquage de pertes de paquets en voix sur ip," in *XXIIe colloque GRETSI*, Sep 2009.

[54] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Communication*, vol. 9, no. 5-6, pp. 453 – 467, 1990. Neuropeech '89.

[55] C. Hamon, "Procédé et dispositif de synthèse de la parole par addition-recouvrement de formes d'onde," 1989.

[56] P. Taylor, R. Caley, A. W. Black, and S. King, "Edinburgh speech tools library," tech. rep., UK Physical Science and Engineering Research Council, 1999.

[57] ITU-T study group 12, "Perceptual evaluation of speech quality (pesq) : An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Tech. Rep. P.862, International Telecommunication Union, Feb 2001.

[58] ITU-T study group 12, "Methods for subjective determination of transmission quality," tech. rep., International Telecommunication Union, Aug 1996.

[59] P. Vary and R. Martin, *Digital Speech Transmission*. Wiley, 2005.

[60] ITU-T study group 12, "Application guide for objective quality measurement based on recommendations p.862, p.862.1 and p.862.2," Tech. Rep. P.862, International Telecommunication Union, Feb 2001.

[61] A. Toutios, U. Musti, S. Ouni, V. Colotte, B. Wrobel Dautcourt, and M.-O. Berger, "Towards a True Acoustic-Visual Speech Synthesis," in *9th International Conference on Auditory-Visual Speech Processing - AVSP2010 AVSP2010*, (Hakone, Kanagawa Japon), pp. POS1–8, 09 2010. ANR - ViSAC - Project N. ANR-08-JCJC-0080-01.

[62] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, (Istanbul , Turkey), pp. 1315–1318, 2000.

[63] E. Gilbert, "Capacity of a burst channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1266, Sep 1960.

[64] ITU-T Users' Group on Software Tools, "Software tools for speech and audio coding standardization," tech. rep., International Telecommunication Union, May 2009.