



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

G.722.2

(01/2002)

SÉRIE G: SYSTÈMES ET SUPPORTS DE
TRANSMISSION, SYSTÈMES ET RÉSEAUX
NUMÉRIQUES

Equipements terminaux numériques – Codage des
signaux analogiques par des méthodes autres que la MIC

**Codage vocal en large bande à 16 kbit/s environ
par codage adaptatif multidébit à large bande
(AMR-WB)**

Recommandation UIT-T G.722.2

RECOMMANDATIONS UIT-T DE LA SÉRIE G
SYSTÈMES ET SUPPORTS DE TRANSMISSION, SYSTÈMES ET RÉSEAUX NUMÉRIQUES

CONNEXIONS ET CIRCUITS TÉLÉPHONIQUES INTERNATIONAUX	G.100–G.199
CARACTÉRISTIQUES GÉNÉRALES COMMUNES À TOUS LES SYSTÈMES ANALOGIQUES À COURANTS PORTEURS	G.200–G.299
CARACTÉRISTIQUES INDIVIDUELLES DES SYSTÈMES TÉLÉPHONIQUES INTERNATIONAUX À COURANTS PORTEURS SUR LIGNES MÉTALLIQUES	G.300–G.399
CARACTÉRISTIQUES GÉNÉRALES DES SYSTÈMES TÉLÉPHONIQUES INTERNATIONAUX HERTZIENS OU À SATELLITES ET INTERCONNEXION AVEC LES SYSTÈMES SUR LIGNES MÉTALLIQUES	G.400–G.449
COORDINATION DE LA RADIOTÉLÉPHONIE ET DE LA TÉLÉPHONIE SUR LIGNES	G.450–G.499
EQUIPEMENTS DE TEST	G.500–G.599
CARACTÉRISTIQUES DES SUPPORTS DE TRANSMISSION	G.600–G.699
EQUIPEMENTS TERMINAUX NUMÉRIQUES	G.700–G.799
Généralités	G.700–G.709
Codage des signaux analogiques en modulation par impulsions et codage	G.710–G.719
Codage des signaux analogiques par des méthodes autres que la MIC	G.720–G.729
Principales caractéristiques des équipements de multiplexage primaires	G.730–G.739
Principales caractéristiques des équipements de multiplexage de deuxième ordre	G.740–G.749
Caractéristiques principales des équipements de multiplexage d'ordre plus élevé	G.750–G.759
Caractéristiques principales des équipements de transcodage et de multiplication numérique	G.760–G.769
Fonctionnalités de gestion, d'exploitation et de maintenance des équipements de transmission	G.770–G.779
Caractéristiques principales des équipements de multiplexage en hiérarchie numérique synchrone	G.780–G.789
Autres équipements terminaux	G.790–G.799
RÉSEAUX NUMÉRIQUES	G.800–G.899
SECTIONS NUMÉRIQUES ET SYSTÈMES DE LIGNES NUMÉRIQUES	G.900–G.999

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

Recommandation UIT-T G.722.2

Codage vocal en large bande à 16 kbit/s environ par codage adaptatif multidébit à large bande (AMR-WB)

Résumé

La présente Recommandation décrit le codeur et le décodeur d'adaptation de haute qualité à des débits multiples en large bande (AMR-WB), qui sont principalement destinés à traiter des signaux de parole d'une largeur de bande de 7 kHz. L'adaptation AMR-WB fonctionne à une multitude de débits binaires compris entre 6,6 kbit/s et 23,85 kbit/s. Le débit peut être modifié à toute limite de trame de 20 ms.

L'annexe C de la présente Recommandation contient un paquetage logiciel intégré en code source C qui permet l'implémentation du codec conforme à la présente Recommandation avec ses Annexes A et B et son Appendice I. Son Annexe D offre aux développeurs un ensemble de vecteurs numériques d'essai formant un outil de vérification et d'indication du fait que ce codec a été correctement implémenté.

Le codec G.722-2 AMR-WB est le même que le codec 3GPP AMR-WB. Les spécifications 3GPP correspondantes sont les normes TS 26.190 pour le codec de parole et TS 26.194 pour le détecteur d'activité vocale.

Source

La Recommandation UIT-T G.722.2, élaborée par la Commission d'études 16 (2001-2004) de l'UIT-T, a été approuvée le 13 janvier 2002 selon la procédure définie dans la Résolution n° 1 de l'AMNT.

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT avait été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

© UIT 2002

Tous droits réservés. Aucune partie de cette publication ne peut être reproduite, par quelque procédé que ce soit, sans l'accord écrit préalable de l'UIT.

TABLE DES MATIÈRES

		Page
1	Domaine d'application	1
2	Références normatives.....	2
3	Définitions, symboles et abréviations.....	2
3.1	Définitions	2
3.2	Symboles	4
3.3	Abréviations	7
4	Description générale	8
4.1	Description fonctionnelle des parties audio	8
4.2	Préparation des échantillons vocaux	8
4.3	Principes du codeur de parole adaptatif à des débits multiples en large bande.....	9
4.4	Principes du décodeur de parole adaptatif à des débits multiples	14
4.5	Séquence et importance subjective des paramètres codés.....	17
5	Description fonctionnelle du codeur.....	17
5.1	Prétraitement.....	17
5.2	Analyse par prédiction linéaire et quantification.....	17
5.2.1	Fenêtrage et calcul d'autocorrélation	18
5.2.2	Algorithme de Levinson-Durbin	18
5.2.3	Conversion de la prédiction linéaire (LP) en paires spectrales d'immitance (ISP).....	19
5.2.4	Conversion des paires spectrales d'immitance (ISP) en prédiction linéaire (LP).....	20
5.2.5	Quantification des coefficients de paire ISP	21
5.2.6	Interpolation des paires ISP	23
5.3	Pondération perceptive	23
5.4	Analyse en boucle ouverte du délai tonal.....	23
5.4.1	Mode 6,60 kbit/s.....	24
5.4.2	Modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s	25
5.5	Calcul de la réponse aux impulsions	25
5.6	Calcul du signal cible	26
5.7	Répertoire adaptatif	26
5.8	Répertoire algébrique	28
5.8.1	Structure du répertoire	28
5.8.2	Indexation des impulsions	32
5.8.3	Exploration du répertoire.....	36
5.9	Quantification des gains de répertoire adaptatif et de répertoire fixe	40
5.10	Mise à jour de la mémoire	41

	Page
5.11	Production du gain dans la bande des hautes fréquences 42
6	Description fonctionnelle du décodeur 42
6.1	Décodage et synthèse de parole 42
6.2	Filtrage passe-haut, renormalisation multiplicatrice et interpolation 45
6.3	Bande des hautes fréquences 45
6.3.1	Production de l'excitation à haute fréquence 46
6.3.2	Filtre de prédiction linéaire pour la bande des hautes fréquences 46
6.3.3	Synthèse de la bande des hautes fréquences 47
7	Répartition détaillée des bits du codec adaptatif à multidébit en large bande 47
8	Séquences de retour à l'état d'origine 56
8.1	Description fonctionnelle 56
8.2	Définitions 57
8.3	Retour à l'état d'origine du codeur 57
8.4	Retour à l'état d'origine du décodeur 57
9	Détecteur d'activité vocale (VAD, <i>voice activity detector</i>) 58
9.1	Symboles du détecteur VAD 58
9.1.1	Variables du détecteur VAD 58
9.1.2	Constantes du détecteur VAD 58
9.1.3	Fonctions 60
9.2	Description fonctionnelle 60
9.2.1	Batterie de filtres et calcul des niveaux de sous-bande 61
9.2.2	Détection de fréquence 63
9.2.3	Décision du détecteur VAD 63
10	Bibliographie (informative) 68

Recommandation UIT-T G.722.2

Codage vocal en large bande à 16 kbit/s environ par codage adaptatif multidébit à large bande (AMR-WB)

1 Domaine d'application

La présente Recommandation décrit l'insertion de blocs d'entrée de 320 échantillons vocaux en format MIC uniforme de 16 bits dans des blocs codés de 132, 177, 253, 285, 317, 365, 397, 461 et 477 bits, ainsi que des blocs codés de 132, 177, 253, 285, 317, 365, 397, 461 et 477 bits dans des blocs de sortie de 320 échantillons vocaux reconstruits. La fréquence d'échantillonnage est de 16 000 échantillons par seconde, ce qui se traduit par un débit du train de bits codé de 6,60, 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s. Le schéma de codage est, pour les modes de codage à débits multiples, le codeur dit à prédiction linéaire à excitation par séquence codée à structure algébrique, ci-après appelé codeur ACELP (ACELP, *algebraic code excited linear prediction coder*). Le codeur ACELP adaptatif à des débits multiples en large bande est appelé codeur AMR-WB. Le codec décrit dans la présente Recommandation utilise également un détecteur d'activité vocale (VAD, *voice activity detector*) intégré.

Les applications prévues pour la présente Recommandation sont les suivantes: voix sur IP (VoIP) et Internet, communications mobiles, RTPC, téléphonie large bande sur RNIS, vidéophonie et téléconférence sur RNIS.

En plus de l'algorithme spécifié dans le corps de la Recommandation G.722.2, ses Annexes A et B ainsi que son Appendice I offrent des fonctionnalités supplémentaires qui permettent l'interopérabilité avec les systèmes sans fil GSM et 3GPP. Ces fonctionnalités ont été initialement mises au point pour ces systèmes mais leur usage n'est pas limité aux applications mobiles. Deux autres Annexes, D et E, décrivent respectivement les vecteurs d'essai et la structure de trame. Ces annexes peuvent être implémentées indépendamment de la présente spécification principale, selon les différentes exigences des systèmes déployant l'algorithme AMR-WB:

- l'Annexe A décrit les aspects de bruit de confort à utiliser avec l'algorithme AMR-WB en fonctionnement à débit de source contrôlé. L'implémentation de cette annexe est essentielle pour l'interopérabilité avec les systèmes sans fil GSM et 3GPP;
- l'Annexe B décrit le fonctionnement à débit de source contrôlé avec l'algorithme AMR-WB. L'implémentation de cette annexe est essentielle pour l'interopérabilité des systèmes sans fil GSM et 3GPP;
- l'Annexe D décrit les séquences numériques d'essai qui constituent un outil de vérification donnant une indication de succès lors de l'implémentation du codec AMR-WB;
- l'Annexe E décrit la structure de trame dont l'usage est recommandé avec les différents modes de fonctionnement de l'algorithme AMR-WB;
- l'Appendice I décrit un exemple de solution pour le masquage d'erreur de trames AMR-WB erronées ou perdues.

Afin d'améliorer l'ergonomie, le code ANSI-C a été regroupé dans une seule Annexe C avec la description de bas niveau de toutes ces fonctionnalités. En cas de divergence entre les descriptions figurant dans une des différentes parties de la présente Recommandation et l'implémentation de ces descriptions dans l'Annexe C, c'est celle-ci qui prévaut.

Le paragraphe 8 décrit une procédure spécifique de réinitialisation, dite "retour à l'état d'origine". Il s'agit d'un élément de service utile pour mettre le codec dans un état initial connu (par exemple aux fins des essais). Le paragraphe 9 spécifie le détecteur d'activité vocale (VAD, *voice activity detector*) utilisé dans ce codec ainsi que lors du fonctionnement à débit de source contrôlé (DTX) spécifié dans l'Annexe B.

2 Références normatives

La présente Recommandation se réfère à certaines dispositions des Recommandations UIT-T et textes suivants qui de ce fait en sont partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Toute Recommandation ou tout texte étant sujet à révision, les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, aux versions les plus récentes des références normatives suivantes. La liste des Recommandations de l'UIT-T en vigueur est régulièrement publiée.

- [1] Recommandation UIT-T G.711 (1988), *Modulation par impulsions et codage (MIC) des fréquences vocales*.

3 Définitions, symboles et abréviations

3.1 Définitions

La présente Recommandation définit les termes suivants:

3.1.1 répertoire adaptatif: dictionnaire contenant des vecteurs d'excitation qui sont adaptés à chaque sous-trame. Le répertoire adaptatif est déduit de l'état du filtre à long terme. La valeur de délai peut être considérée comme un pointeur sur le répertoire adaptatif.

3.1.2 répertoire algébrique: dictionnaire fixe dans lequel un code algébrique sert à construire les vecteurs d'excitation (vecteurs d'innovation). L'excitation contient un petit nombre d'impulsions non nulles avec des ensembles entrelacés de positions possibles. Les amplitudes et les positions des impulsions du k^e vecteur de code d'excitation peuvent être déduites de son pointeur k par l'application d'une règle exigeant une mémorisation physique minimale ou nulle, contrairement aux répertoires stochastiques dans lesquels le trajet entre le pointeur et le vecteur de code associé implique des tables d'exploration.

3.1.3 traitement antidispersion: procédure de post-traitement adaptatif appliquée au vecteur du répertoire fixe afin de diminuer les artéfacts perceptuels provoqués par la dispersion d'un vecteur du répertoire fixe.

3.1.4 analyse du délai tonal en boucle fermée: recherche dans le répertoire adaptatif, c'est-à-dire processus d'estimation de la valeur de la hauteur (ou du délai) d'après le signal vocal d'entrée pondéré et l'état du filtre à long terme. Lors de l'exploration en boucle fermée, le délai tonal est recherché au moyen d'une boucle à minimisation d'erreur (analyse et synthèse). Dans le codec adaptatif multidébit large bande, la recherche tonale en boucle fermée est effectuée pour chaque sous-trame.

3.1.5 coefficients de forme directe: un des formats de mémorisation des paramètres de filtre à court terme. Dans le codec adaptatif multidébit large bande, tous les filtres utilisés pour modifier des échantillons de parole utilisent des coefficients de forme directe.

3.1.6 répertoire fixe: répertoire qui contient des vecteurs d'excitation pour les filtres de synthèse de parole. Le contenu de ce répertoire n'est pas adaptatif (c'est-à-dire qu'il est fixe). Dans le codec adaptatif multidébit large bande, le répertoire fixe est mis en œuvre au moyen d'un répertoire algébrique.

3.1.7 délais fractionnaires: ensemble de délais dont la résolution se situe au niveau des sous-échantillons, auquel le codec adaptatif multidébit large bande utilise une résolution de 1/4 ou 1/2 échantillon.

3.1.8 trame: intervalle de temps égal à 20 ms (soit 320 échantillons à la fréquence d'échantillonnage de 16 kHz).

3.1.9 fréquences spectrales d'impédance: (voir paire spectrale d'impédance)

3.1.10 paire spectrale d'impédance: transformée des paramètres du codage LPC. Les paires spectrales d'impédance sont obtenues par décomposition de la fonction de transfert du filtre inversé $A(z)$ en un ensemble de deux fonctions de transfert, l'une ayant une symétrie paire et l'autre ayant une symétrie impaire. Les paires spectrales d'impédance (également nommées fréquences spectrales d'impédance) sont les racines de ces polynômes sur le cercle unitaire z .

3.1.11 délais entiers: Ensembles de valeurs de délai ayant la résolution d'un échantillon entier.

3.1.12 filtre d'interpolation: filtre à réponse impulsionnelle finie (FIR) servant à produire une estimation d'échantillons à la résolution des sous-échantillons, compte tenu d'une entrée échantillonnée à la résolution d'un échantillon entier. Dans la présente implémentation, le filtre d'interpolation possède les caractéristiques d'un filtre passe-bas. Le répertoire adaptatif est donc le résultat de l'interpolation par filtrage passe-bas des excitations antérieures.

3.1.13 filtre inverse: ce filtre supprime la corrélation à court terme issue du signal vocal. Il modélise une réponse fréquentielle inverse du tractus aérien.

3.1.14 délai: retard du filtre à long terme. Il s'agit normalement de la période fondamentale vraie ou d'un de ses multiples ou sous-multiples.

3.1.15 fenêtre d'analyse par prédiction linéaire (LP): les coefficients du filtre à court terme sont calculés pour chaque trame au moyen des échantillons vocaux filtrés en passe-haut à l'intérieur de la fenêtre d'analyse. Dans le codec adaptatif multidébit large bande, la longueur de la fenêtre d'analyse est toujours de 384 échantillons. Dans tous les modes, une seule fenêtre asymétrique sert à produire un seul ensemble de coefficients LP. La préexploration de 5 ms est utilisée dans cette analyse.

3.1.16 coefficients LP: terme générique décrivant les coefficients de prédiction linéaire [LP, également appelés coefficients de codage prédictif linéaire (LPC)] du filtre à court terme.

3.1.17 mode: utilisé isolément, ce terme désigne le mode du codec de source, c'est-à-dire le mode de l'un des codecs de source employés dans le codec AMR-WB.

3.1.18 recherche (de hauteur) tonale en boucle ouverte: processus d'estimation directe du plus proche délai optimal à partir de l'entrée vocale pondérée, afin de simplifier l'analyse tonale et de limiter la recherche de hauteur tonale en boucle fermée à un petit nombre de délais autour des délais estimés en boucle ouverte. Dans le codec adaptatif multidébit large bande, une recherche tonale en boucle ouverte est effectuée sur une sous-trame sur deux.

3.1.19 signal résiduel: signal de sortie résultant d'une opération de filtrage inverse.

3.1.20 filtre de synthèse à court terme: filtre qui introduit dans le signal d'excitation une corrélation à court terme qui modélise la réponse impulsionnelle du tractus aérien.

3.1.21 filtre de pondération perceptive: ce filtre est utilisé dans la recherche par analyse et synthèse des répertoires. Ce filtre exploite les propriétés des formants en terme de masquage du bruit (résonances du tractus aérien) par moindre pondération des erreurs dans les régions proches des fréquences du formant et par plus grande pondération des erreurs dans les régions éloignées de ces fréquences.

3.1.22 sous-trame: intervalle de temps égal à 5 ms (soit 80 échantillons à la fréquence d'échantillonnage de 16 kHz).

3.1.23 quantification vectorielle: méthode de regroupement de plusieurs paramètres dans un vecteur afin de les quantifier simultanément.

3.1.24 réponse d'entrée nulle: signal de sortie d'un filtre résultant d'entrées antérieures, c'est-à-dire résultant de l'état actuel du filtre compte tenu de l'application d'une entrée de zéros.

3.1.25 réponse d'état nul: signal de sortie d'un filtre résultant de l'entrée actuelle, sans application d'entrées antérieures, c'est-à-dire compte tenu du fait que les informations d'état sont toutes des zéros dans le filtre.

3.2 Symboles

La présente Recommandation utilise les symboles suivants:

$A(z)$	Filtre inverse avec coefficients non quantifiés
$\hat{A}(z)$	Filtre inverse avec coefficients quantifiés
$H(z) = \frac{1}{\hat{A}(z)}$	Filtre de synthèse vocale avec coefficients quantifiés
a_i	Paramètres non quantifiés de prédiction linéaire (coefficients de forme directe)
\hat{a}_i	Paramètres quantifiés de prédiction linéaire
m	Ordre du modèle LP
$W(z)$	Filtre de pondération perceptive (coefficients non quantifiés)
γ_1	Facteur de pondération perceptive
T	Délai tonal entier le plus proche du délai tonal fractionnaire en boucle fermée de la sous-trame
β	Coefficient du préfiltre adaptatif (le gain tonal quantifié)
$H_{h1}(z)$	Filtre passe-haut de prétraitement
$w(n)$	Fenêtre d'analyse LP
L_1	Longueur de la première partie de la fenêtre d'analyse LP $w(n)$
L_2	Longueur de la deuxième partie de la fenêtre d'analyse LP $w(n)$
$r(k)$	Autocorrélations du signal vocal cadré dans la fenêtre $s'(n)$
$w_{lag}(i)$	Fenêtre de délai pour les autocorrélations (extension de largeur de bande de 60 Hz)
f_0	Extension de largeur de bande en Hz
f_s	Fréquence d'échantillonnage en Hz
$r'(k)$	Autocorrélations modifiées (après extension de largeur de bande)
$E(i)$	Erreur de prédiction dans la i ème itération de l'algorithme de Levinson
k_i	i ème coefficient de réflexion
$a_j^{(i)}$	j ème coefficient de forme directe dans la i ème itération de l'algorithme de Levinson
$F_1'(z)$	Polynôme ISF symétrique
$F_2'(z)$	Polynôme ISF antisymétrique
$F_1(z)$	Polynôme $F_1'(z)$
$F_2(z)$	Polynôme $F_2'(z)$ avec racines $z = 1$ et $z = -1$ éliminées
q_i	Paires spectrales d'immitance (ISP, <i>immitance spectral pair</i>) dans le domaine cosinusoidal
\mathbf{q}	Vecteur ISP dans le domaine cosinusoidal

$\hat{\mathbf{q}}_i^{(n)}$	Vecteur quantifié ISP à la <i>i</i> ème sous-trame de la trame <i>n</i>
ω_i	Fréquences spectrales d'immitance (ISF)
$T_m(x)$	Polynôme de Chebyshev du <i>mième</i> ordre
$f_1(i), f_2(i)$	Coefficients des polynômes $F_1(z)$ et $F_2(z)$
$f'_1(i), f'_2(i)$	Coefficients des polynômes $F'_1(z)$ et $F'_2(z)$
$f(i)$	Coefficients soit de $F_1(z)$ soit de $F_2(z)$
$C(x)$	Polynôme de sommation du polynôme de Chebyshev
x	Cosinus de fréquence angulaire ω
λ_k	Coefficients de récurrence pour l'évaluation du polynôme de Chebyshev
f_i	Fréquences spectrales d'immitance (ISF) en Hz
$\mathbf{f}^t = [f_1 f_2 \dots f_{16}]$	Représentation vectorielle des paires ISF en Hz
$\mathbf{z}(n)$	Vecteur ISF à moyenne supprimée à la trame <i>n</i>
$\mathbf{r}(n)$	Vecteur résiduel de prédiction ISF à la trame <i>n</i>
$\mathbf{p}(n)$	Vecteur ISF prédit à la trame <i>n</i>
$\hat{\mathbf{r}}(n-1)$	Vecteur résiduel quantifié à la trame antérieure
$\hat{\mathbf{r}}_i^k$	Sous-vecteur ISF quantifié <i>i</i> à l'indice de quantification <i>k</i>
d_i	Distance entre les fréquences spectrales d'immitance f_{i+1} et f_{i-1}
$h(n)$	Réponse impulsionnelle du filtre pondéré de synthèse
$H(z)W(z)$	Filtre pondéré de synthèse
T_1	Entier le plus proche du délai tonal fractionnaire de la précédente (1 ^{ère} ou 3 ^{ème}) sous-trame
$s'(n)$	Signal vocal cadré dans la fenêtre
$s_w(n)$	Signal vocal pondéré
$\hat{s}(n)$	Signal vocal reconstruit
$x(n)$	Signal cible pour recherche dans le répertoire adaptatif
$x_2(n), \mathbf{x}_2^t$	Signal cible pour recherche dans le répertoire algébrique
$res_{LP}(n)$	Signal LP résiduel
$c(n)$	Vecteur du répertoire fixe
$v(n)$	Vecteur du répertoire adaptatif
$y(n) = v(n)*h(n)$	Vecteur filtré du répertoire adaptatif
$y_k(n)$	Excitation filtrée antérieure
$u(n)$	Signal d'excitation
$\hat{u}'(n)$	Signal d'excitation normalisé par le gain et mis en évidence
T_{op}	Meilleur délai en boucle ouverte

t_{min}	Valeur de recherche du délai minimal
t_{max}	Valeur de recherche du délai maximal
$R(k)$	Terme de corrélation à maximiser lors de la recherche dans le répertoire adaptatif
$R(k)_t$	Valeur interpolée de $R(k)$ pour le délai entier k et la fraction t
A_k	Terme de corrélation à maximiser lors de la recherche dans le répertoire algébrique à l'indice k
C_k	Corrélation dans le numérateur A_k à l'indice k
E_{Dk}	Energie dans le dénominateur of A_k à l'indice k
$\mathbf{d} = \mathbf{H}^t \mathbf{x}_2$	Corrélation entre le signal cible $x_2(n)$ et la réponse impulsionnelle $h(n)$, c'est-à-dire la cible filtrée en inverse
\mathbf{H}	Partie triangulaire inférieure de la matrice convolutionnelle de Töpliz avec diagonale $h(0)$ et diagonales inférieures $h(1), \dots, h(63)$
$\Phi = \mathbf{H}^t \mathbf{H}$	Matrice de corrélations de $h(n)$
$d(n)$	Éléments du vecteur \mathbf{d}
$\phi(i, j)$	Éléments de la matrice symétrique Φ
\mathbf{c}_k	Vecteur d'innovation
C	Corrélation dans le numérateur de A_k
m_i	Position de la i ème impulsion
ϑ_i	Amplitude de la i ème impulsion
N_p	Nombre d'impulsions dans l'excitation par le répertoire fixe
E_D	Energie dans le dénominateur de A_k
$res_{LTP}(n)$	Signal résiduel de prédiction à long terme normalisé
$b(n)$	Signal utilisé pour préréglage les signes lors de la recherche dans le répertoire algébrique
$s_b(n)$	Signal de signe pour la recherche dans le répertoire algébrique
$d'(n)$	Cible filtrée en inverse avec extension au signe
$\phi'(i, j)$	Éléments modifiés de la matrice Φ , y compris les informations de signe
$\mathbf{z}^t, z(n)$	Vecteur du répertoire fixe convolué avec $h(n)$
$E(n)$	Energie d'innovation à moyenne supprimée (en dB)
\bar{E}	Moyenne de l'énergie d'innovation
$\tilde{E}(n)$	Energie prédite
$[b_1 \ b_2 \ b_3 \ b_4]$	Coefficients de prédiction en moyenne mobile
$\hat{R}(k)$	Erreur quantifiée de prédiction à la sous-trame k
E_I	Energie moyenne d'innovation
$R(n)$	Erreur de prédiction de la quantification du gain du répertoire fixe

E_Q	Erreur de quantification du gain du répertoire fixe
$e(n)$	Etats du filtre de synthèse $1/\hat{A}(z)$
$e_w(n)$	Erreur à pondération perceptuelle de la recherche par analyse et synthèse
η	Facteur de normalisation par le gain pour l'excitation accentuée
g_c	Gain du répertoire fixe
g'_c	Gain prédit du répertoire fixe
\hat{g}_c	Gain quantifié du répertoire fixe
g_p	Gain du répertoire adaptatif
\hat{g}_p	Gain quantifié du répertoire adaptatif
$\gamma_{gc} = g_c/g'_c$	Facteur de correction entre le gain g_c et le gain estimé g'_c
$\hat{\gamma}_{gc}$	Valeur optimale pour $\hat{\gamma}_{gc}$
γ_{sc}	Facteur de normalisation par le gain

3.3 Abréviations

La présente Recommandation utilise les abréviations suivantes:

ACELP	prédiction linéaire à excitation par séquence codée à structure algébrique (<i>algebraic code excited linear prediction</i>)
AGC	commande adaptative de gain (<i>adaptive gain control</i>)
AMR	multidébit adaptatif (<i>adaptive multirate</i>)
AMR-WB	multidébit adaptatif large bande (<i>adaptive multirate wideband</i>)
CELP	prédiction linéaire avec excitation par (code) (<i>code excited linear prediction</i>)
FIR	réponse impulsionnelle finie (<i>finite impulse response</i>)
ISF	fréquence spectrale d'immitance (<i>immittance spectral frequency</i>)
ISP	paire spectrale d'immitance (<i>immittance spectral pair</i>)
ISPP	permutation entrelacée d'impulsion isolée (<i>interleaved single-pulse permutation</i>)
LP	prédiction linéaire (<i>linear prediction</i>)
LPC	codage linéaire prédictif (<i>linear predictive coding</i>)
LTP	prédicteur à long terme (ou prédiction à long terme) (<i>long-term predictor</i>) (<i>or long-term prediction</i>)
MA	moyenne mobile (<i>moving average</i>)
S-MSVQ	quantification vectorielle en plusieurs phases après subdivision (<i>split-multistage vector quantization</i>)
WB	largeur de bande (<i>wideband</i>)

4 Description générale

La présente Recommandation est structurée comme suit:

- le paragraphe 4.1 contient une description fonctionnelle des parties audio, y compris les fonctions A/D et D/A;

- le paragraphe 4.2 décrit le format d'entrée pour le codeur AMR-WB et le format de sortie pour le décodeur AMR-WB;
- les paragraphes 4.3 et 4.4 présentent respectivement une description simplifiée des principes du processus de codage et de décodage AMR-WB;
- le paragraphe 4.5 indique la séquence et l'importance subjective des paramètres codés.
- le paragraphe 5 contient la description fonctionnelle du codec AMR-WB;
- le paragraphe 6 décrit les procédures de décodage;
- le paragraphe 7 présente sous forme de tableau la répartition détaillée des bits du codec AMR-WB;
- le paragraphe 8 décrit l'opération de retour à l'état d'origine.

4.1 Description fonctionnelle des parties audio

La conversion analogique-numérique et numérique-analogique comportera en principe les éléments suivants:

- 1) conversion d'analogique en MIC numérique uniforme [1]
 - microphone;
 - dispositif de réglage du niveau d'entrée;
 - filtre anticrênelage d'entrée;
 - dispositif d'échantillonnage à 16 kHz et mémorisation;
 - conversion d'analogique en numérique uniforme avec représentation par 14 bits.

Le format uniforme doit être représenté en complément à deux.

- 2) Conversion de MIC numérique uniforme en analogique [1]
 - conversion de MIC numérique uniforme 14 bits/16 kHz en analogique;
 - dispositif de mémorisation;
 - filtre de reconstruction y compris la correction $x/\sin(x)$;
 - dispositif de réglage du niveau de sortie;
 - écouteur ou haut-parleur.

Dans l'équipement terminal, la fonction A/D peut être réalisée

- par conversion directe en format MIC uniforme de 14 bits.

Pour l'opération D/A, les opérations inverses sont effectuées.

4.2 Préparation des échantillons vocaux

Le codeur est alimenté en données contenant des échantillons de 14 bits en mots de 16 bits justifiés à gauche. Le décodeur produit des données dans le même format. A l'extérieur du codec vocal, un traitement complémentaire doit être appliqué si les données de trafic se présentent dans un format différent.

4.3 Principes du codeur de parole adaptatif à des débits multiples en large bande

Le codec AMR-WB se compose de neuf codecs de source avec des débits de 23,85, 23,05, 19,85, 18,25, 15,85, 14,25, 12,65, 8,85 et 6,60 kbit/s.

Ce codec est fondé sur le modèle de codage prédictif linéaire avec excitation par code (CELP, *code excited linear prediction*). Le signal d'entrée est préaccentué au moyen du filtre $H_{\text{pre-emph}}(z) = 1 - \mu z^{-1}$. Le modèle CELP est appliqué ensuite au signal préaccentué. Un filtre de synthèse à prédiction linéaire (LP) du 16ème ordre (ou à court terme) est utilisé, ce qui est donné par:

$$H(z) = \frac{1}{\hat{A}(z)} = \frac{1}{1 + \sum_{i=1}^m \hat{a}_i z^{-i}} \quad (1)$$

où $\hat{a}_i, i = 1, \dots, m$ sont les paramètres de prédiction linéaire (LP, *linear prediction*) et où $m = 16$ est l'ordre du prédicteur. Le filtre de synthèse à long terme (ou tonal) est habituellement donné par:

$$\frac{1}{B(z)} = \frac{1}{1 - g_p z^{-T}} \quad (2)$$

où T est le délai tonal et où g_p est le gain tonal. Le filtre de synthèse tonale est implémenté au moyen de la méthode dite du répertoire adaptatif.

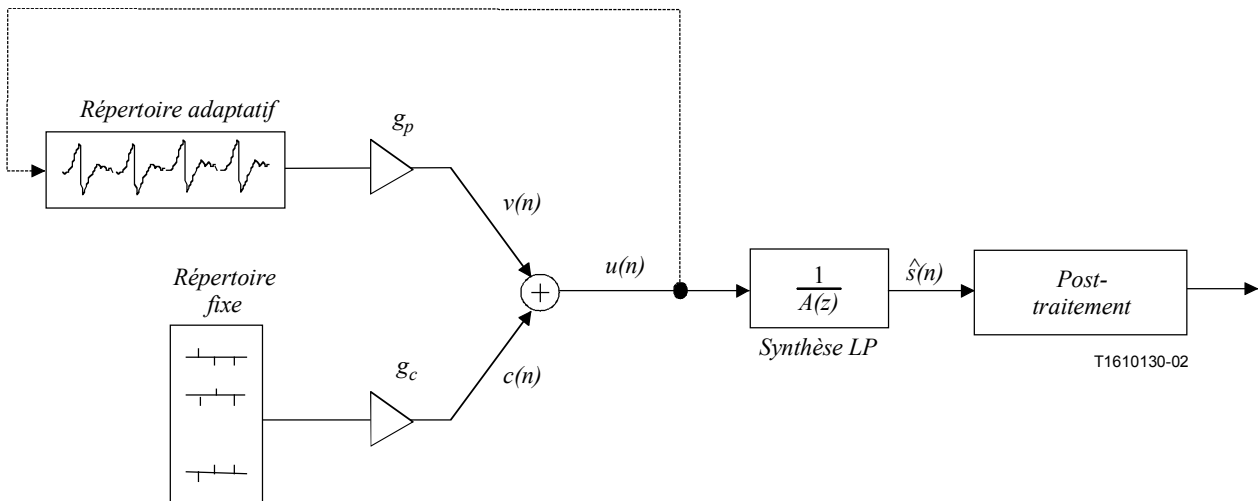


Figure 1/G.722.2 – Schéma fonctionnel simplifié du modèle de synthèse CELP

Le modèle de synthèse vocale par prédiction CELP est représenté dans la Figure 1. Dans ce modèle, le signal d'excitation injecté à l'entrée du filtre de synthèse LP à court terme est construit par adjonction de deux vecteurs d'excitation issus des répertoires adaptatif et fixe (d'innovation). La parole est synthétisée par injection des deux vecteurs convenablement choisis dans ces répertoires par le filtre de synthèse à court terme. La séquence d'excitation optimale est choisie dans un répertoire au moyen d'une procédure de recherche par analyse et synthèse dans laquelle l'erreur entre le signal vocal original et le signal vocal synthétisé est minimisée en fonction d'une valeur mesurée de distorsion à pondération perceptive.

Le filtre à pondération perceptive utilisé dans la technique de recherche par analyse et synthèse est donné par la relation suivante:

$$W(z) = A(z / \gamma_1) H_{\text{de-emph}}(z) \quad (3)$$

où $A(z)$ est le filtre de prédiction linéaire non quantifiée, $H_{\text{de-emph}} = \frac{1}{1 - 0,68z^{-1}}$, et $\gamma_1 = 0,92$ est le facteur de pondération perceptive. Le filtre de pondération utilise les paramètres de prédiction linéaire non quantifiée.

Le codeur effectue l'analyse des paramètres de codage LPC, de prédiction LTP et de répertoire fixe à la fréquence d'échantillonnage de 12,8 kHz. Le codeur travaille sur des trames vocales de 20 ms. A chaque trame, le signal vocal est analysé afin d'extraire les paramètres du modèle de prédiction CELP (coefficients de filtre LP, indices et gains de répertoire adaptatif et de répertoire fixe). En plus de ces paramètres, des indices de gain dans la bande des hautes fréquences sont calculés en mode 23,85 kbit/s. Ces paramètres sont codés et transmis. Dans le décodeur, ces paramètres sont

décodés et les signaux vocaux sont synthétisés par filtrage du signal d'excitation reconstruit au moyen du filtre de synthèse LP.

Le flux des signaux dans le codeur est décrit par la Figure 2. Après décimation, l'on effectue un filtrage passe-haut et un filtrage de préaccentuation. L'analyse de prédiction linéaire est effectuée trame par trame. L'ensemble des paramètres LP est converti en paires spectrales d'immitance (ISP) et quantifié vectoriellement par quantification vectorielle en plusieurs phases après subdivision (S-MSVQ, *split-multistage vector quantization*). La trame vocale est subdivisée en 4 sous-trames de 5 ms chacune (64 échantillons à la fréquence d'échantillonnage de 12,8 kHz). Les paramètres de répertoire adaptatif et de répertoire fixe sont transmis sous-trame par sous-trame. Selon la sous-trame, l'on utilise les paramètres LP quantifiés et non quantifiés, ou leur version interpolée. Un délai tonal en boucle ouverte est estimé dans une sous-trame sur deux en fonction du signal vocal à pondération perceptive.

Les opérations suivantes sont ensuite répétées pour chaque sous-trame:

- le signal cible $x(n)$ est calculé par filtrage du signal résiduel LP dans le filtre pondéré de synthèse $W(z)H(z)$ avec mise à jour des états initiaux des filtres par filtrage de l'erreur entre signal LP résiduel et excitation (ce qui est équivalent à la méthode courante consistant à soustraire, du signal vocal pondéré, la réponse d'entrée nulle du filtre pondéré de synthèse);
- la réponse impulsionnelle, $h(n)$ du filtre pondéré de synthèse est calculée;
- l'analyse tonale en boucle fermée est ensuite effectuée (afin de trouver le délai tonal et le gain), au moyen de la cible $x(n)$ et de la réponse impulsionnelle $h(n)$, par recherche autour du délai tonal en boucle ouverte. L'on utilise le délai tonal fractionnaire avec résolution au quart ou à la moitié d'échantillon (selon le mode et la valeur du délai tonal). Lors de la recherche du délai tonal fractionnaire, le filtre d'interpolation a une réponse en fréquence passe-bas. Par ailleurs, il y a deux caractéristiques passe-bas possibles dans le répertoire adaptatif: cette information est codée sur 1 fanion;
- le signal cible $x(n)$ est mis à jour par suppression de la contribution du répertoire adaptatif (vecteur de code adaptatif filtré) et cette nouvelle cible, $x_2(n)$, est utilisée lors de la recherche dans le répertoire algébrique fixe (afin de trouver l'innovation optimale);
- les gains des répertoires adaptatif et fixe sont quantifiés vectoriellement sur 6 ou 7 bits (avec moyenne mobile (MA, *moving average*) de prédiction appliquée au gain du répertoire fixe);
- finalement, les mémoires du filtre sont mises à jour (au moyen du signal d'excitation déterminé) afin de trouver le signal cible dans la sous-trame suivante.

La répartition des bits des modes du codec AMR-WB est présentée dans le Tableau 1. Dans chaque trame vocale de 20 ms, 132, 177, 253, 285, 317, 365, 397, 461 et 477 bits sont produits, correspondant à un débit de 6,60, 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s. Une répartition des bits plus détaillée entre les paramètres du codec est indiquée dans les Tableaux 12a-12i. Noter que les bits de plus fort poids (MSB, *most significant bit*) sont toujours envoyés en premier.

**Tableau 1/G.722.2 – Répartition des bits de l'algorithme de codage AMR-WB
pour trame de 20 ms**

Mode	Paramètre	1^{ère} sous-trame	2^e sous-trame	3^e sous-trame	4^e sous-trame	Total par trame	
23,85 kbit/s	Fanion VAD					1	
	ISP					46	
	Filtrage LTP	1	1	1	1	4	
	Délai tonal	9	6	9	6	30	
	Code algébrique	88	88	88	88	352	
	Gain de répertoire	7	7	7	7	28	
	Energie HB	4	4	4	4	16	
	Total						477
23,05 kbit/s	Fanion VAD					1	
	ISP					46	
	Filtrage LTP	1	1	1	1	4	
	Délai tonal	9	6	9	6	30	
	Code algébrique	88	88	88	88	352	
	Gains	7	7	7	7	28	
	Total						461
	19,85 kbit/s	Fanion VAD					1
ISP						46	
Filtrage LTP		1	1	1	1	4	
Délai tonal		9	6	9	6	30	
Code algébrique		72	72	72	72	288	
Gain de répertoire		7	7	7	7	28	
Total							397
18,25 kbit/s		Fanion VAD					1
	ISP					46	
	Filtrage LTP	1	1	1	1	4	
	Délai tonal	9	6	9	6	30	
	Code algébrique	64	64	64	64	256	
	Gains	7	7	7	7	28	
	Total						365
	15,85 kbit/s	Fanion VAD					1
ISP						46	
Filtrage LTP		1	1	1	1	4	
Délai tonal		9	6	9	6	30	
Code algébrique		52	52	52	52	208	
Gains		7	7	7	7	28	
Total							317

**Tableau 1/G.722.2 – Répartition des bits de l'algorithme de codage AMR-WB
pour trame de 20 ms**

Mode	Paramètre	1^{ère} sous-trame	2^e sous-trame	3^e sous-trame	4^e sous-trame	Total par trame
14,25 kbit/s	Fanion VAD					1
	ISP					46
	Filtrage LTP	1	1	1	1	4
	Délai tonal	9	6	9	6	30
	Code algébrique	44	44	44	44	176
	Gains	7	7	7	7	28
	Total					
12,65 kbit/s	Fanion VAD					1
	ISP					46
	Filtrage LTP	1	1	1	1	4
	Délai tonal	9	6	9	6	30
	Code algébrique	36	36	36	36	144
	Gains	7	7	7	7	28
	Total					
8,85 kbit/s	Fanion VAD					1
	ISP					46
	Délai tonal	8	5	8	5	26
	Code algébrique	20	20	20	20	80
	Gains	6	6	6	6	24
	Total					
6,60 kbit/s	Fanion VAD					1
	ISP					36
	Délai tonal	8	5	5	5	23
	Code algébrique	12	12	12	12	48
	Gains	6	6	6	6	24
	Total					

4.4 Principes du décodeur de parole adaptatif à des débits multiples

Le flux de signaux dans le décodeur est décrit par la Figure 3. Dans le décodeur, les indices transmis sont extraits du train binaire reçu puis sont décodés afin d'obtenir les paramètres de codeur dans chaque trame d'émission. Ces paramètres sont le vecteur de paires ISP, les 4 délais tonaux fractionnaires, les 4 paramètres de filtrage LTP, les 4 vecteurs de code d'innovation et les 4 ensembles de gains tonaux et d'innovation quantifiés vectoriellement. En mode 23,85 kbit/s, l'indice de gain est décodé. Le vecteur de paire ISP est converti en coefficients du filtre LP et interpolé afin d'obtenir les filtres LP à chaque sous-trame. Puis, dans chacune des sous-frames d'une longueur de 64 échantillons:

- l'excitation est construite par addition des vecteurs de code adaptatifs et innovatifs, normalisés par leur gain respectif;

- le signal vocal de 12,8 kHz est reconstruit par filtrage de l'excitation dans le filtre de synthèse LP;
- le signal vocal reconstruit est désaccentué.

Finalement, le signal vocal reconstruit est suréchantillonné à 16 kHz et le signal vocal dans la bande des hautes fréquences est ajouté à la bande de fréquence allant de 6 kHz à 7 kHz.

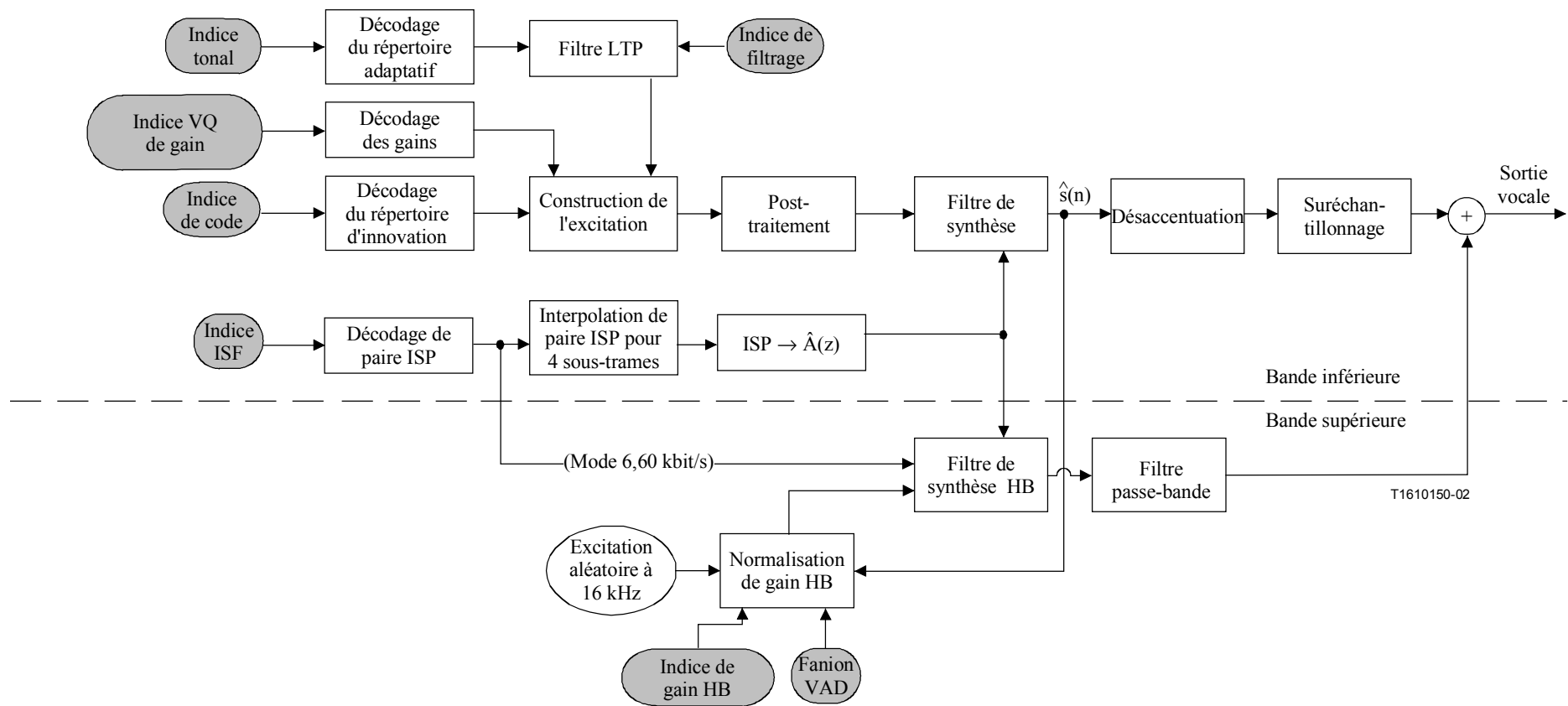


Figure 3 /G.722.2 – Schéma fonctionnel détaillé du décodeur ACELP

4.5 Séquence et importance subjective des paramètres codés

Le codeur produit les informations de sortie dans un format et selon une séquence uniques. Le décodeur doit recevoir les mêmes informations de la même façon. Les Tableaux 12a-12i montrent la séquence des bits de sortie et leur répartition pour chaque paramètre.

Les différents paramètres du signal vocal codé et leurs bits individuels ont une importance inégale en terme de qualité subjective.

5 Description fonctionnelle du codeur

Le présent paragraphe décrit les différentes fonctions du codeur représenté sur la Figure 2.

5.1 Prétraitement

Le codeur effectue l'analyse des paramètres de codage LPC, de prédiction LTP et du répertoire fixe à la fréquence d'échantillonnage de 12,8 kHz. Le signal d'entrée doit donc subir une décimation de 16 kHz à 12,8 kHz. Cette décimation est effectuée comme suit: un quadruple suréchantillonnage est d'abord appliqué, puis le signal de sortie est injecté dans un filtre passe-bas à réponse FIR $H_{decim}(z)$ dont la fréquence de coupure est réglée à 6,4 kHz. Le signal subit ensuite un quintuple sous-échantillonnage. Le délai de filtrage est compensé par l'insertion de zéros à la fin du vecteur d'entrée.

Après la décimation et avant le processus de codage, deux fonctions de prétraitement sont appliquées au signal: un filtrage passe-haut et une préaccentuation (avec renormalisation réductrice du signal).

(La renormalisation réductrice consiste à diviser l'entrée par un facteur 2 afin de diminuer la probabilité de débordements dans l'implémentation en virgule fixe.)

Le filtre passe-haut sert de protection contre des composants parasites à basse fréquence. L'on utilise un filtre à fréquence de coupure de 50 Hz, dont la formule est la suivante:

$$H_{h1}(z) = \frac{0,989502 - 1,979004z^{-1} + 0,989502z^{-2}}{1 - 1,978882z^{-1} + 0,9799126z^{-2}} \quad (4)$$

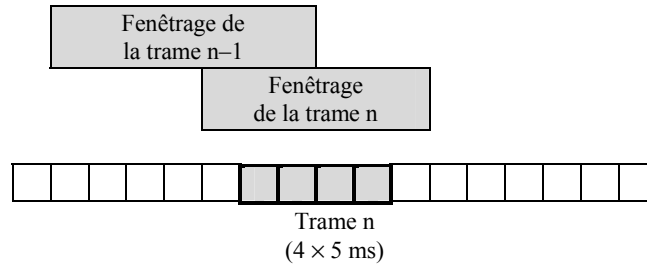
[L'on combine la renormalisation réductrice avec le filtrage passe-haut en divisant par 2 les coefficients figurant au numérateur de la fraction $H_{h1}(z)$.]

Lors de la préaccentuation, un filtre passe-haut du premier ordre est utilisé afin d'accentuer les très hautes fréquences. Ce filtre est donné par:

$$H_{pre-emph}(z) = 1 - 0,68z^{-1} \quad (5)$$

5.2 Analyse par prédiction linéaire et quantification

Une analyse par prédiction à court terme (LP) est effectuée à chaque trame vocale au moyen de la méthode d'autocorrélation dans une fenêtre asymétrique de 30 ms. Une redondance de 5 ms est utilisée lors du calcul d'autocorrélation. La structure de trame est décrite ci-dessous.



Les autocorrélations de signaux vocaux fenêtrés sont converties en coefficients de prédiction linéaire au moyen de l'algorithme de Levinson-Durbin. Les coefficients LP sont ensuite transformés en paires du domaine ISP aux fins de la quantification et de l'interpolation. Les filtres interpolés, quantifiés et non quantifiés, sont reconvertis en coefficients de filtre LP (afin de construire les filtres de synthèse et de pondération à chaque sous-trame).

5.2.1 Fenêtrage et calcul d'autocorrélation

L'analyse LP est effectuée à chaque trame dans une fenêtre asymétrique, dont le poids est concentré sur la quatrième sous-trame et se compose de deux parties: la première est une fenêtre de Hamming et la deuxième est un quart de cycle d'une fonction de Hamming cosinusoidale. Cette fenêtre est donnée par:

$$\begin{aligned}
 w(n) &= 0,54 - 0,46 \cos\left(\frac{2\pi n}{2L_1 - 1}\right), & n = 0, \dots, L_1 - 1 \\
 &= \cos\left(\frac{2\pi(n - L_1)}{4L_2 - 1}\right), & n = L_1, \dots, L_1 + L_2 - 1
 \end{aligned} \tag{6}$$

où les valeurs $L_1=256$ et $L_2=128$ sont utilisées.

Les autocorrélations du signal vocal cadré dans la fenêtre $s'(n), n = 0, \dots, 383$ sont calculées par:

$$r(k) = \sum_{n=k}^{383} s'(n) s'(n-k), \quad k = 0, \dots, 16 \tag{7}$$

et une extension de largeur de bande de 60 Hz est utilisée par retardement du fenêtrage des autocorrélations au moyen de la fenêtre [3]

$$w_{lag}(i) = \exp\left[-\frac{1}{2} \left(\frac{2\pi f_0 i}{f_s}\right)^2\right], \quad i = 1, \dots, 16 \tag{8}$$

où $f_0 = 60$ Hz est l'extension de largeur de bande et $f_s = 12\,800$ Hz est la fréquence d'échantillonnage. Par ailleurs, $r(0)$ est multiplié par le facteur de correction du bruit blanc égal à 1,0001, ce qui est équivalent à l'adjonction d'un bruit de fond à -40 dB.

5.2.2 Algorithme de Levinson-Durbin

Les autocorrélations modifiées $r'(0) = 1,0001r(0)$ et $r'(k) = r(k)w_{lag}(k), k = 1, \dots, 16$, sont utilisées afin d'obtenir les coefficients du filtre LP $a_k, k = 1, \dots, 16$ au moyen de la résolution de l'ensemble d'équations suivant:

$$\sum_{k=1}^{16} a_k r'(|i-k|) = -r'(i), \quad i = 1, \dots, 16 \tag{9}$$

L'ensemble d'équations (9) est résolu au moyen de l'algorithme de Levinson-Durbin [3], qui utilise la récurrence suivante:

$$E(0) = r'(0)$$

Pour $i = 1$ à 16 faire

$$k_i = -\left[r'(i) + \sum_{j=1}^{i-1} a_j^{i-1} r'(i-j) \right] / E(i-1)$$

$$a_i^{(i)} = k_i$$

Pour $j = 1$ à $i-1$ faire

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}$$

$$E(i) = (1 - k_i^2) E(i-1)$$

La solution finale est donnée sous la forme $a_j = a_j^{(16)}$, $j = 1, \dots, 16$.

Les coefficients du filtre LP sont convertis en représentation de paires ISP [5] aux fins de la quantification et de l'interpolation. Les conversions vers le domaine des paires ISP et inversement vers le domaine du filtre sont décrites aux § 5.2.3 et 5.2.4.

5.2.3 Conversion de la prédiction linéaire (LP) en paires spectrales d'impédance (ISP)

Les coefficients du filtre LP a_k , $k = 1, \dots, 16$, sont convertis en représentation de paires ISP aux fins de la quantification et de l'interpolation. Pour un filtre LP du 16^e ordre, les paires ISP sont définies comme étant les racines des polynômes de somme et de différence suivants:

$$f_1'(z) = A(z) + z^{-16} A(z^{-1}) \quad (10)$$

et

$$f_2'(z) = A(z) - z^{-16} A(z^{-1}) \quad (11)$$

respectivement. (Les polynômes $f_1'(z)$ et $f_2'(z)$ sont respectivement symétrique et antisymétrique). L'on peut démontrer que toutes les racines de ces polynômes sont sur le cercle des unités et qu'elles alternent l'une avec l'autre [6]. Le polynôme $f_2'(z)$ a deux racines, à $z = 1$ ($\omega = 0$) et à $z = -1$ ($\omega = \pi$). Pour éliminer ces deux racines, l'on définit les nouveaux polynômes suivants:

$$f_1(z) = f_1'(z) \quad (12)$$

et

$$f_2(z) = f_2'(z) / (1 - z^{-2}) \quad (13)$$

Les polynômes $f_1(z)$ et $f_2(z)$ ont respectivement 8 et 7 racines conjuguées sur le cercle des unités ($e^{\pm j\omega_i}$). Donc, ces polynômes peuvent être écrits comme suit:

$$F_1(z) = (1 + a[16]) \prod_{i=0,2,\dots,14} (1 - 2q_i z^{-1} + z^{-2}) \quad (14)$$

et

$$F_2(z) = (1 + a[16]) \prod_{i=1,3,\dots,13} (1 - 2q_i z^{-1} + z^{-2}) \quad (15)$$

où $q_i = \cos(\omega_i)$ avec ω_i qui représente les fréquences spectrales d'impédance (ISF) et où a [16] est le dernier coefficient du prédicteur. Les paires ISF répondent à la relation d'ordonnement

$0 < \omega_1 < \omega_2 < \dots < \omega_{16} < \pi$. L'on considère q_i comme étant les paires ISP dans le domaine cosinusoidal.

Étant donné que les deux polynômes $f_1(z)$ et $f_2(z)$ sont symétriques, seuls les 8 et 7 premiers coefficients de chaque polynôme, respectivement, et le dernier coefficient du prédicteur, ont besoin d'être calculés.

Les coefficients de ces polynômes sont trouvés par les relations récursives

pour $i = 0$ à 7

$$f_1(i) = a_i + a_{m-i}$$

$$f_2(i) = a_i - a_{m-i} + f_2(i-2) \quad (16)$$

$$f_1(8) = 2a_8$$

où $m = 16$ est l'ordre du prédicteur, et $f_2(-2) = f_2(-1) = 0$.

Les paires ISP sont trouvées par évaluation des polynômes $F_1(z)$ et $F_2(z)$ à 100 points équidistants entre 0 et π et par vérification des changements de signe. Un changement de signe signifie l'existence d'une racine et l'intervalle du changement de signe est ensuite divisé 4 fois afin de mieux localiser la racine. Le polynôme de Chebyshev est utilisé pour évaluer $F_1(z)$ et $F_2(z)$ [7]. Dans cette méthode, les racines sont trouvées directement dans le domaine cosinusoidal $\{q_i\}$. Les polynômes $F_1(z)$ et $F_2(z)$ évalués à $z = e^{j\omega}$ peuvent être écrits sous la forme:

$$F_1(\omega) = 2e^{-j8\omega}C_1(x) \text{ et } F_2(\omega) = 2e^{-j7\omega}C_2(x) \quad (17)$$

avec

$$C_1(x) = \sum_{i=0}^7 f_1(i)T_{8-i}(x) + f_1(8)/2, \text{ et } C_2(x) = \sum_{i=0}^6 f_2(i)T_{8-i}(x) + f_2(7)/2 \quad (18)$$

où $T_m = \cos(m\omega)$ est le polynôme de Chebyshev du *mième* ordre, et où $f(i)$ sont les coefficients soit de $F_1(z)$ ou de $F_2(z)$, calculés au moyen des équations de (16). Le polynôme $C(x)$ est évalué à une certaine valeur de $x = \cos(\omega)$ au moyen de la relation récursive:

pour $k = n_f - 1$ en décroissance jusqu'à 1

$$b_k = 2xb_{k+1} - b_{k+2} + f(n_f - k)$$

fin

$$C(x) = xb_1 - b_2 + f(n_f)/2$$

où $n_f = 8$ si $C_1(x)$ et $n_f = 7$ si $C_2(x)$, avec les valeurs initiales $b_{n_f} = f(0)$ et $b_{n_f+1} = 0$. Les détails de la méthode d'évaluation du polynôme de Chebyshev sont donnés en [7].

5.2.4 Conversion des paires spectrales d'immitance (ISP) en prédiction linéaire (LP)

Une fois que les paires ISP sont quantifiées et interpolées, elles sont reconverties vers le domaine des coefficients LP $\{a_k\}$. La conversion vers le Domaine LP est effectuée comme suit. Les coefficients de $F_1(z)$ et $F_2(z)$ sont trouvés par expansion des équations (14) et (15) sur la base des paires ISP quantifiées et interpolées $q_i = , i = 0, \dots, m-1$, où $m = 16$. La relation récursive suivante est utilisée pour calculer $f_1(z)$

pour $i = 2$ à $m/2$

$$f_1(i) = -2q_{2i-2}f_1(i-1) + 2f_1(i-2)$$

pour $j = i-1$ à 2

$$f_1(j) = f_1(j) - 2q_{2i-2}f_1(j-1) + f_1(j-2)$$

fin

$$f_1(1) = f_1(1) - 2q_{2i-2}$$

fin

avec les valeurs initiales $f_1(0) = 1$ et $f_1(1) = -2q_0$. Les coefficients $f_2(i)$ sont calculés de la même façon par remplacement de q_{2i-2} par q_{2i-1} et de $m/2$ par $m/2-1$, avec les conditions initiales $f_2(0) = 1$ et $f_2(1) = -2q_1$.

Une fois que les coefficients $f_1(z)$ et $f_2(z)$ sont trouvés, $F_2(z)$ est multiplié par $1-z^{-2}$, afin d'obtenir $F'_2(z)$; soit

$$\begin{aligned} f'_2(i) &= f_2(i) - f_2(i-2), & i = 2, \dots, m/2-1 \\ f'_1(i) &= f_1(i) & i = 0, \dots, m/2 \end{aligned} \quad (19)$$

Ensuite $F'_1(z)$ et $F'_2(z)$ sont multipliés par $1+q_{m-1}$ et $1-q_{m-1}$, respectivement. Soit

$$\begin{aligned} f'_2(i) &= (1-q_{m-1})f'_2(i), & i = 0, \dots, m/2-1 \\ f'_1(i) &= (1+q_{m-1})f'_1(i) & i = 0, \dots, m/2 \end{aligned}$$

Finalement les coefficients LP sont trouvés par

$$\begin{aligned} a_i &= 0.5f'_1(i) + 0.5f'_2(i), & i = 1, \dots, m/2-1 \\ &0.5f'_1(i) - 0.5f'_2(i), & i = m/2+1, \dots, m-1 \\ &0.5f'_1(m/2), & i = m/2 \\ &q_{m-1}, & i = m \end{aligned} \quad (20)$$

Ce résultat est directement déduit de la relation $A(z) = (F'_1(z) + F'_2(z))/2$, compte tenu du fait que $F'_1(z)$ et $F'_2(z)$ sont respectivement des polynômes symétrique et antisymétrique.

5.2.5 Quantification des coefficients de paire ISP

Les coefficients du filtre LP sont quantifiés au moyen de la représentation des paires ISP dans le domaine fréquentiel; soit

$$\begin{aligned} f_i &= \frac{f_s}{2\pi} \arccos(q_i), & i = 0, \dots, 14 \\ &= \frac{f_s}{4\pi} \arccos(q_i), & i = 15 \end{aligned} \quad (21)$$

où f_i sont les paires ISF en Hz [0,6400] et où $f_s = 12\,800$ est la fréquence d'échantillonnage. Le vecteur de fréquence ISF est donné par $\mathbf{r}^t = [f_0, f_1, \dots, f_{15}]$, avec t indiquant une transposition.

Une prédiction du premier ordre à moyenne mobile (MA) est appliquée et le vecteur de fréquence ISF résiduel est quantifié au moyen d'une combinaison de quantification vectorielle après subdivision (SVQ, *split vector quantization*) et d'une quantification vectorielle en plusieurs phases après subdivision (MSVQ). La prédiction et la quantification sont effectuées comme suit. Soit $\mathbf{z}(n)$ le vecteur ISF à moyenne supprimée à la trame n . Le vecteur résiduel de prédiction $\mathbf{r}(n)$ est donné par:

$$\mathbf{r}(n) = \mathbf{z}(n) - \mathbf{p}(n) \quad (22)$$

où $\mathbf{p}(n)$ est le vecteur LSF prédit à la trame n . Une prédiction du premier ordre à moyenne mobile (MA) est utilisée, dans laquelle:

$$\mathbf{p}(n) = \frac{1}{3} \hat{\mathbf{r}}(n-1) \quad (23)$$

où $\hat{\mathbf{r}}(n-1)$ est le vecteur résiduel quantifié à la trame antérieure.

Le vecteur résiduel ISF \mathbf{r} subit une quantification vectorielle en plusieurs phases après subdivision (S-MSVQ). Le vecteur est subdivisé en 2 sous-vecteurs $\mathbf{r}_1(n)$ et $\mathbf{r}_2(n)$ de dimensions 9 et 7, respectivement. Ces 2 sous-vecteurs sont quantifiés en deux phases. Dans la première phase, le vecteur $\mathbf{r}_1(n)$ est quantifié avec 8 bits et le vecteur $\mathbf{r}_2(n)$ avec 8 bits.

Pour les modes à 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s, les vecteurs d'erreur de quantification $\mathbf{r}_i^{(2)} = \mathbf{r} - \hat{\mathbf{r}}_i, i = 1, 2$ sont respectivement subdivisés dans la phase suivante en 3 et 2 sous-vecteurs. Ces sous-vecteurs sont quantifiés au moyen des débits décrits dans le Tableau 2.

Tableau 2/G.722.2 – Quantification de vecteur ISP pour les modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s

1. Vecteur ISP non quantifié d'une longueur de 16 éléments				
2. Phase 1 (\mathbf{r}_1) 8 bits			2. Phase 1 (\mathbf{r}_2) 8 bits	
3. Phase 2 ($\mathbf{r}^{(2)}$) _{1,0-2} 6 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{1,3-5} 7 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{1,6-8} 7 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{2,0-2} 5 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{2,3-6} 5 bits

Pour le mode 6,60 kbit/s, les vecteurs d'erreur de quantification $\mathbf{r}_i^{(2)} = \mathbf{r} - \hat{\mathbf{r}}_i, i = 1, 2$ sont respectivement subdivisés dans la phase suivante en 2 et 1 sous-vecteurs. Les sous-vecteurs sont quantifiés au moyen des débits décrits dans le Tableau 3.

Tableau 3/G.722.2 – Quantification de vecteur ISP pour le mode 6,60 kbit/s

1. Vecteur ISP non quantifié d'une longueur de 16 éléments		
2. PHASE 1 (\mathbf{r}_1) 8 bits		2. PHASE 1 (\mathbf{r}_2) 8 bits
3. Phase 2 ($\mathbf{r}^{(2)}$) _{1,0-4} 7 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{1,5-8} 7 bits	3. Phase 2 ($\mathbf{r}^{(2)}$) _{2,0-6} 6 bits

Une mesure de distorsion de paire ISP par erreur quadratique est utilisée dans le processus de quantification. En général, pour un sous-vecteur résiduel d'entrée ISP ou d'erreur $\mathbf{r}_i, i = 1, 2$ et pour un vecteur quantifié à l'indice $k, \hat{\mathbf{r}}_i^k$, la quantification est effectuée par découverte de l'indice k qui minimise

$$E = \sum_{i=m}^n \left[r_i - \hat{r}_i^k \right]^2 \quad (24)$$

où m et n sont les premier et dernier éléments du sous-vecteur.

5.2.6 Interpolation des paires ISP

L'ensemble de paramètres LP quantifiés (et non quantifiés) est utilisé pour la quatrième sous-trame tandis que les première, deuxième, et troisième sous-trames utilisent une interpolation linéaire des paramètres dans les trames adjacentes. L'interpolation est effectuée sur les paires ISP dans le domaine \mathbf{q} . Soit $\hat{\mathbf{q}}_4^{(n)}$ le vecteur ISP à la 4^e sous-trame de la trame, et $\hat{\mathbf{q}}_4^{(n-1)}$ le vecteur ISP à la 4^e sous-trame de la trame antérieure $n-1$. Les vecteurs interpolés ISP aux 1^{ère}, 2^e et 3^e sous-trames sont donnés par

$$\begin{aligned}\hat{\mathbf{q}}_1^{(n)} &= 0,55\hat{\mathbf{q}}_4^{(n-1)} + 0,45\hat{\mathbf{q}}_4^{(n)} \\ \hat{\mathbf{q}}_2^{(n)} &= 0,2\hat{\mathbf{q}}_4^{(n-1)} + 0,8\hat{\mathbf{q}}_4^{(n)} \\ \hat{\mathbf{q}}_3^{(n)} &= 0,04\hat{\mathbf{q}}_4^{(n-1)} + 0,96\hat{\mathbf{q}}_4^{(n)}\end{aligned}$$

La même formule est utilisée pour l'interpolation des paires ISP non quantifiées. Les vecteurs ISP interpolés sont utilisés pour calculer un filtre LP différent à chaque sous-trame (quantifiées aussi bien que non quantifiées) au moyen de la méthode de conversion de paires ISP en vecteurs LP qui est décrite au § 5.2.4.

5.3 Pondération perceptive

Le filtre de pondération perceptive traditionnel $W(z) = A(z/\gamma_1)/A(z/\gamma_2)$ possède des limitations inhérentes dans la modélisation simultanée de la structure du formant et de l'écart requis du niveau spectral à court terme. L'écart de niveau spectral à court terme est plus prononcé dans les signaux en bande large en raison de la large étendue dynamique entre basses et hautes fréquences. Une solution à ce problème consiste à introduire le filtre de préaccentuation à l'entrée, à calculer le filtre LP $A(z)$ sur la base du signal vocal préaccentué $s(n)$, et à utiliser un filtre $W(z)$ modifié par réglage de son dénominateur. Cette structure découple nettement la pondération par le formant par rapport à l'écart de niveau à court terme.

Un filtre de pondération de la forme $W(z) = A(z/\gamma_1)H_{\text{de-emph}}(z)$ est utilisé, où

$$H_{\text{de-emph}} = \frac{1}{1 - \beta_1 z^{-1}} \text{ et } \beta_1 = 0,68.$$

Etant donné que $A(z)$ est calculé sur la base du signal vocal préaccentué $s(n)$, l'écart de niveau à court terme du filtre $1/A(z/\gamma_1)$ est moins prononcé par rapport au cas où $A(z)$ est calculé sur la base du signal vocal original. Etant donné que la désaccentuation est effectuée du côté décodeur, on peut montrer que le spectre d'erreur de quantification est conformé par un filtre having a fonction de transfert $W^{-1}(z)H_{\text{de-emph}}(z) = 1/A(z/\gamma_1)$. Ainsi, le spectre de l'erreur de quantification est conformé par un filtre dont la fonction de transfert est $1/A(z/\gamma_1)$, avec le terme $A(z)$ calculé sur la base du signal vocal préaccentué.

5.4 Analyse en boucle ouverte du délai tonal

Selon le mode, l'analyse du délai tonal en boucle ouverte est effectuée une fois par trame (de 10 ms chacune) ou deux fois par trame (de 10 ms chacune) afin de trouver deux estimations du délai tonal dans chaque trame. Cela est effectué afin de simplifier l'analyse tonale et de limiter la recherche en boucle fermée du délai tonal à un petit nombre de délais autour des délais en boucle ouverte estimés.

L'estimation du délai tonal en boucle ouverte est fondée sur le signal vocal pondéré $s_w(n)$ qui est obtenu par filtrage du signal vocal d'entrée dans le filtre de pondération

$W(z) = A(z/\gamma_1)H_{\text{de-emph}}(z)$, où $H_{\text{de-emph}} = \frac{1}{1-\beta_1 z^{-1}}$ et $\beta_1 = 0,68$. Dans une sous-trame de longueur L , le signal vocal pondéré est donc donné par :

$$s_w(n) = s(n) + \sum_{i=1}^{16} a_i \gamma_1^i s(n-i) + \beta_1 s_w(n-1), \quad n = 0, \dots, L-1 \quad (25)$$

L'analyse du délai tonal en boucle ouverte est effectuée sur un signal décimé par deux. Le signal décimé est obtenu par filtrage $s_w(n)$ dans un filtre à réponse FIR du 4^e ordre, $H_{\text{decim2}}(z)$ puis par sous-échantillonnage de la sortie d'un facteur deux afin d'obtenir le signal $s_{wd}(n)$.

5.4.1 Mode 6,60 kbit/s

L'analyse du délai tonal en boucle ouverte est effectuée une fois par trame (toutes les 20 ms) afin de trouver une estimation du délai tonal dans chaque trame.

L'analyse du délai tonal en boucle ouverte est effectuée comme suit. Tout d'abord, la corrélation du signal vocal décimé et pondéré est déterminée comme suit pour chaque valeur d du délai tonal:

$$C(d) = \sum_{n=0}^{128} S_{wd}(n) s_{wd}(n-d) w(d), \quad d = 17, \dots, 115 \quad (26)$$

où $w(d)$ est une fonction de pondération. Le délai tonal estimé est le délai qui maximise la fonction de corrélation pondérée $C(d)$. La pondération accentue les valeurs inférieures du délai tonal, ce qui réduit la probabilité de sélectionner un multiple du délai correct. La fonction de pondération se compose de deux parties: une fonction d'accentuation de faible délai tonal, $w_l(d)$, et une fonction d'accentuation adjacente du délai de la trame antérieure, $w_n(d)$:

$$w(d) = w_l(d) w_n(d) \quad (27)$$

La fonction d'accentuation de faible délai tonal est donnée par:

$$w_l(d) = c w(d) \quad (28)$$

où $c w(d)$ est défini par une table contenue dans la description du calcul en virgule fixe. La fonction d'accentuation adjacente du délai de la trame antérieure dépend du délai tonal des trames vocales antérieures:

$$w_n(d) = \begin{cases} c w(|T_{old} - d| + 98), & v > 0,8 \\ 1,0, & \text{sinon} \end{cases} \quad (29)$$

où T_{old} est le délai tonal filtré médian des 5 demi-trames vocales antérieurement voisées et où v est un paramètre adaptatif. Si la trame est classifiée comme étant voisée parce que son gain en boucle ouverte est $g > 0,6$, alors la valeur v est fixée à 1,0 pour la trame suivante. Sinon, la valeur v est mise à jour par $v = 0,9v$. Le gain en boucle ouverte est donné par:

$$g = \frac{\sum_{n=0}^{127} s_{wd}(n) s_{wd}(n-d_{max})}{\sqrt{\sum_{n=0}^{127} s_{wd}^2(n) \sum_{n=0}^{127} s_{wd}^2(n-d_{max})}} \quad (30)$$

où d_{max} est le délai tonal qui maximise $C(d)$. Le filtre médian n'est mis à jour que pendant les trames vocales voisées. La pondération dépend de la fiabilité des précédents délais tonaux. Si des trames antérieures contenaient des signaux vocaux non voisés ou du silence, la pondération est affaiblie par le paramètre v .

5.4.2 Modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s

L'analyse du délai tonal en boucle ouverte est effectuée deux fois par trame (toutes les 10 ms) afin de trouver deux estimations du délai tonal dans chaque trame.

L'analyse du délai tonal en boucle ouverte est effectuée comme suit. Tout d'abord, la corrélation du signal vocal décimé et pondéré est déterminée pour chaque valeur d du délai tonal comme suit:

$$C(d) = \sum_{n=0}^{63} s_{wd}(n)s_{wd}(n-d)w(d), \quad d = 17, \dots, 115 \quad (31)$$

où $w(d)$ est une fonction de pondération. Le délai tonal estimé est le délai qui maximise la fonction de corrélation pondérée $C(d)$. La pondération accentue les valeurs inférieures du délai tonal afin de réduire la probabilité de sélection d'un multiple du délai correct. La fonction de pondération se compose de deux parties: une fonction d'accentuation de faible délai tonal, $w_l(d)$, et une fonction d'accentuation adjacente du délai de la trame antérieure, $w_n(d)$:

$$w(d) = w_l(d)w_n(d) \quad (32)$$

La fonction d'accentuation de faible délai tonal est donnée par:

$$w_l(d) = cw(d) \quad (33)$$

où $cw(d)$ est défini par une table contenue dans la description du calcul en virgule fixe. La fonction d'accentuation adjacente du délai de la trame antérieure dépend du délai tonal des trames vocales antérieures:

$$w_n(d) = \begin{cases} cw(|T_{old} - d| + 98), & v > 0.8, \\ 1,0, & \text{sinon} \end{cases} \quad (34)$$

où T_{old} est le délai tonal filtré médian des 5 demi-trames vocales antérieurement voisées et où v est un paramètre adaptatif. Si la trame est classifiée comme étant voisée parce que son gain en boucle ouverte est $g > 0,6$, alors la valeur v est fixée à 1,0 pour la trame suivante. Sinon, la valeur v est mise à jour par $v = 0,9v$. Le gain en boucle ouverte est donné par:

$$g = \frac{\sum_{n=0}^{63} s_{wd}(n)s_{wd}(n-d_{max})}{\sqrt{\sum_{n=0}^{63} s_{wd}^2(n) \sum_{n=0}^{63} s_{wd}^2(n-d_{max})}} \quad (35)$$

où d_{max} est le délai tonal qui maximise $C(d)$. Le filtre médian n'est mis à jour que pendant les trames vocales voisées. La pondération dépend de la fiabilité des précédents délais tonaux. Si des trames antérieures contenaient des signaux vocaux non voisés ou du silence, la pondération est affaiblie par le paramètre v .

5.5 Calcul de la réponse aux impulsions

La réponse impulsionnelle, $h(n)$, du filtre pondéré de synthèse $H(z)W(z) = A(z/\gamma_1)H_{de-emph}(z)/\hat{A}(z)$ est calculée à chaque sous-trame. Cette réponse impulsionnelle est requise pour la recherche dans les répertoires adaptatif et fixe. La réponse impulsionnelle $h(n)$ est calculée par filtrage du vecteur des coefficients du filtre $A(z/\gamma_1)$ étendu par des zéros dans les deux filtres $1/\hat{A}(z)$ et $H_{de-emph}(z)$.

5.6 Calcul du signal cible

Le signal cible pour recherche dans le répertoire adaptatif est habituellement calculé par soustraction de la réponse à entrée nulle du filtre pondéré de synthèse $H(z)W(z) = A(z/\gamma_1)H_{\text{de-emph}}(z)/\hat{A}(z)$ du signal vocal pondéré $s_w(n)$. Cette soustraction est effectuée sous-trame par sous-trame.

Une procédure équivalente pour calculer le signal cible, qui est utilisée dans ce codec, est le filtrage du signal LP résiduel $r(n)$ dans la combinaison du filtre de synthèse $1/\hat{A}(z)$ et du filtre de pondération $A(z/\gamma_1)H_{\text{de-emph}}(z)$. Après détermination de l'excitation pour la sous-trame, les états initiaux de ces filtres sont mis à jour par filtrage de la différence entre le signal résiduel LP et l'excitation. La mise à jour des mémoires de ces filtres est expliquée au § 5.10.

Le signal résiduel $r(n)$ qui est requis pour trouver le vecteur cible est également utilisé lors de la recherche dans le répertoire adaptatif afin d'étendre le tampon d'excitation antérieure. Cela simplifie la procédure de recherche dans le répertoire adaptatif pour les délais inférieurs à la longueur de sous-trame de 64 positions, comme cela sera expliqué dans le paragraphe 5.7. Le signal résiduel LP est donné par

$$r(n) = s(n) - \sum_{i=1}^{16} \hat{a}_i s(n-i), \quad n = 0, \dots, 63 \quad (36)$$

5.7 Répertoire adaptatif

La recherche dans le répertoire adaptatif est effectuée sous-trame par sous-trame. Elle consiste à effectuer la recherche en boucle fermée du délai tonal, puis à calculer le vecteur de code adaptatif par interpolation de l'excitation antérieure au délai tonal fractionnaire choisi.

Les paramètres du répertoire adaptatif (ou paramètres tonaux) sont le délai et le gain du filtre tonal. Dans la phase de recherche phase, l'excitation est étendue par le signal résiduel LP afin de simplifier la recherche en boucle fermée.

Dans les modes 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s, dans les première et troisième sous-trames, un délai tonal fractionnaire est utilisé avec résolutions $1/4$ dans l'étendue $[34, 127\frac{3}{4}]$, résolutions $1/2$ dans l'étendue $[128, 159\frac{1}{2}]$, et entiers seulement dans l'étendue $[160, 231]$. Pour les deuxième et quatrième sous-trames, une résolution tonale de $1/4$ est toujours utilisée dans l'étendue $[T_1-8, T_1+7\frac{3}{4}]$, où T_1 est l'entier le plus proche du délai tonal fractionnaire de la précédente (1^{ère} ou 3^e) sous-trame.

En mode 8,85 kbit/s, dans les première et troisième sous-trames, un délai tonal fractionnaire est utilisé avec résolutions $1/2$ dans l'étendue $[34, 91\frac{1}{2}]$, et entiers seulement dans l'étendue $[92, 231]$. Pour les deuxième et quatrième sous-trames, une résolution tonale de $1/2$ est toujours utilisée dans l'étendue $[T_1-8, T_1+7\frac{1}{2}]$, où T_1 est l'entier le plus proche du délai tonal fractionnaire de la précédente (1^{ère} ou 3^e) sous-trame.

En mode 6,60 kbit/s, dans la première sous-trame, un délai tonal fractionnaire est utilisé avec résolutions $1/2$ dans l'étendue $[34, 91\frac{1}{2}]$, et entiers seulement dans l'étendue $[92, 231]$. Pour les deuxième, troisième et quatrième sous-trames, une résolution tonale de $1/2$ est toujours utilisée dans

l'étendue $[T_1-8, T_1+7\frac{1}{2}]$, où T_1 est l'entier le plus proche du délai tonal fractionnaire de la première sous-trame.

L'analyse tonale en boucle fermée est effectuée autour de l'estimation du délai tonal en boucle ouverte, sous-trame par sous-trame. Dans les modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s, l'étendue $T_{op} \pm 7$, limitée par 34...231, est recherchée dans les première (et troisième) sous-trames. En mode 6,60 kbit/s mode, dans la première sous-trame, l'étendue $T_{op} \pm 7$, limitée par 34...231, est recherchée. Pour tous les modes et pour les autres sous-trames, l'analyse tonale en boucle fermée est effectuée autour du délai tonal entier sélectionné dans la sous-trame précédente, comme décrit ci-dessus. Dans les modes 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s, le délai tonal est codé avec 9 bits dans les première et troisième sous-trames et le délai relatif des autres sous-trames est codé avec 6 bits. En mode 8,85 kbit/s, le délai tonal est codé avec 8 bits dans les première et troisième sous-trames et le délai relatif des autres sous-trames est codé avec 5 bits. En mode 6,60 kbit/s mode, le délai tonal est codé avec 8 bits dans la première sous-trame et le délai relatif des autres sous-trames est codé avec 5 bits.

La recherche en boucle fermée du délai tonal est effectuée par minimisation de l'erreur quadratique pondérée entre le signal vocal original et le signal vocal. Cela est réalisé par maximisation du terme:

$$T_k = \frac{\sum_{n=0}^{63} x(n)y_k(n)}{\sqrt{\sum_{n=0}^{63} y_k(n)y_k(n)}} \quad (37)$$

où $x(n)$ est le signal cible et où $y_k(n)$ est l'excitation antérieure filtrée au délai k [excitation antérieure convoluée avec $h(n)$]. Noter que la recherche étendue est limitée autour du délai tonal en boucle ouverte comme expliqué plus haut.

La convolution $y_k(n)$ est calculée pour le premier délai dans la recherche étendue; pour les autres délais, elle est mise à jour au moyen de la relation récursive:

$$y_k(n) = y_{k-1}(n-1) + u(-k)h(n) \quad (38)$$

où $u(n), n = -(231+17), \dots, 63$, est le tampon d'excitation. Noter que, dans la phase de recherche, les échantillons $u(n), n = 0, \dots, 63$, ne sont pas connus mais sont requis pour les délais tonaux inférieurs à 64. Afin de simplifier la recherche, le signal résiduel LP est copié dans $u(n)$ afin de rendre la relation de l'équation (38) valide pour tous les délais.

Une fois que la valeur d'entier optimale du délai tonal est déterminée, les fractions par rapport à $-\frac{3}{4}$ à $\frac{3}{4}$ avec échelon de $\frac{1}{4}$ autour de cet entier sont essayées. La recherche du délai tonal fractionnaire est effectuée par interpolation de la corrélation normalisée dans l'équation (37) et par recherche de son maximum. Une fois que le délai tonal fractionnaire est déterminé, le terme $v'(n)$ est calculé par interpolation du signal d'excitation antérieur $u(n)$ à la phase (fraction) indiquée. (L'interpolation est effectuée au moyen de deux filtres à réponse FIR (fonctions $\sin(x)/x$ mises en fenêtre de Hamming); le premier pour interpoler le terme dans l'équation 34 avec la fonction $\sin(x)/x$ tronquée à ± 17 et le second pour interpoler l'excitation antérieure avec la fonction $\sin(x)/x$ tronquée à ± 63). Les filtres ont leur fréquence de coupure (-3 dB) à 6 000 Hz dans le domaine suréchantillonné, ce qui signifie que les filtres d'interpolation offrent une réponse en fréquence passe-bas. Ainsi, même lorsque le délai tonal est une valeur d'entier, l'excitation par répertoire adaptatif se compose d'une version filtrée en passe-bas de l'excitation antérieure au délai indiqué, et non pas d'une copie directe de ce signal. Par ailleurs, pour les délais plus petits que la longueur de sous-trame, l'excitation par répertoire adaptatif est effectuée sur la base de l'excitation antérieure après interpolation et filtrage passe-bas et non par répétition de cette excitation antérieure.

Afin d'améliorer la qualité de prédiction tonale dans les signaux en bande large, un prédicteur tonal dépendant de la fréquence est utilisé. Cela est important dans les signaux en bande large étant donné que la périodicité ne s'étend pas nécessairement sur tout le spectre. Dans cet algorithme, il y a deux trajets de signal associés aux ensembles respectifs de paramètres de répertoire tonal, où chaque trajet de signal contient un dispositif de calcul d'erreur de prédiction tonale d'un vecteur de code tonal par rapport à un dispositif de recherche dans un répertoire tonal. Un de ces deux trajets se compose d'un filtre passe-bas pour filtrage du vecteur de code tonal. L'erreur de prédiction tonale est calculée pour ces deux trajets de signal. Le trajet de signal ayant la plus petite valeur calculée d'erreur de prédiction tonale est sélectionné, avec le gain tonal associé.

Le filtre passe-bas utilisé dans le deuxième trajet est de la forme $B_{LP}(z) = 0,18z + 0,64 + 0,18z^{-1}$. Noter que 1 bit est utilisé afin de coder le trajet choisi.

Ainsi, pour les modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/, il y a deux possibilités afin de produire le répertoire adaptatif $v(n)$, $v(n) = v'(n)$ dans le premier trajet, ou

$v(n) = \sum_{i=1}^1 b_{LP}(i+1)v'(n+i)$ dans le deuxième trajet, où $\mathbf{b}_{LP} = [0,18, 0,64, 0,18]$. Le trajet qui se

traduit par l'énergie minimale du signal cible $x_2(n)$ défini dans l'équation 40 est sélectionné pour le vecteur filtré du répertoire adaptatif. Pour le mode 6,60 kbit/s, $v(n)$ est toujours

$$v(n) = \sum_{i=1}^1 b_{LP}(i+1)v'(n+i)$$

Le gain du répertoire adaptatif est ensuite trouvé par:

$$g_p = \frac{\sum_{n=0}^{63} x(n)y(n)}{\sqrt{\sum_{n=0}^{63} y(n)y(n)}}, \quad \text{bounded by } 0 \leq g_p \leq 1,2 \quad (39)$$

où $y(n) = v(n)*h(n)$ est le vecteur filtré du répertoire adaptatif (réponse à l'état nul de $H(z)W(z)$ à $v_i(n)$). Afin d'assurer la stabilité, le gain du répertoire adaptatif g_p est limité par 0,95, si les gains du répertoire adaptatif des précédentes sous-frames ont été petits et si les filtres LP de la précédente sous-trame ont été presque instables.

5.8 Répertoire algébrique

5.8.1 Structure du répertoire

La structure du répertoire est fondée sur un modèle de permutation entrelacée d'impulsion isolée (ISPP, *interleaved single-pulse permutation*). Les 64 positions dans le vecteur de code sont divisées en 4 pistes de positions entrelacées, avec 16 positions dans chaque piste. Les différents répertoires aux différents débits sont construits par insertion d'un certain nombre d'impulsions signées dans les pistes (de 1 à 6 impulsions par piste). L'indice de répertoire, ou mot codé, représente les positions et les signes d'impulsion dans chaque piste. Ainsi, aucune mémoire de répertoire n'est requise, étant donné que le vecteur d'excitation dans le décodeur peut être construit au moyen des informations contenues dans l'indice lui-même (pas de tables de consultation).

Une importante caractéristique du répertoire utilisé est qu'il s'agit d'un répertoire dynamique formé d'un répertoire algébrique suivi d'un préfiltre adaptatif $F(z)$ qui renforce les composantes spectrales spéciales afin d'améliorer la qualité vocale de synthèse. Un préfiltre approprié aux signaux en bande large est utilisé, dans lequel $F(z)$ se compose de deux parties: une partie d'amélioration de la périodicité $1/(1-0,85z^{-T})$ et une partie relative à l'écart de niveau à court terme $(1 - \beta_1 z^{-1})$, où T est l'entier part du délai tonal et β_1 est associé à la verbalisation de la précédente sous-trame et est limité par $[0,0, 0,5]$. La recherche dans le répertoire est effectuée dans le domaine algébrique par combinaison du filtre $F(z)$ avec le filtre pondéré de synthèse avant la recherche dans le répertoire.

Ainsi, la réponse impulsionnelle $h(n)$ doit toujours être modifiée de façon à inclure le préfiltre $F(z)$. Soit, $h(n) \leftarrow h(n) * f(n)$.

Les structures de répertoire des différents débits binaires sont donné ci-dessous.

5.8.1.1 Modes 23,85 et 23,05 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 24 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chaque piste contient six impulsions, comme indiqué dans le Tableau 4.

Tableau 4/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 23,85 et 23,05 kbit/s

Piste	Impulsion	Positions
1	$i_0, i_4, i_8, i_{12}, i_{16}, i_{20}$	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	$i_1, i_5, i_9, i_{13}, i_{17}, i_{21}$	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	$i_2, i_6, i_{10}, i_{14}, i_{18}, i_{22}$	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	$i_3, i_7, i_{11}, i_{15}, i_{19}, i_{23}$	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Les six impulsions dans une même piste sont codées avec 22 bits.

Cela donne un total de 88 bits (22+22+22+22) pour le code algébrique.

5.8.1.2 Mode 19,85 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 18 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chacune des deux premières pistes contient cinq impulsions et chacune des autres pistes contient quatre impulsions, comme indiqué dans le Tableau 5.

Tableau 5/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 19,85 kbit/s

Piste	Impulsion	Positions
1	$i_0, i_4, i_8, i_{12}, i_{16}$	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	$i_1, i_5, i_9, i_{13}, i_{17}$	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6, i_{10}, i_{14}	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7, i_{11}, i_{15}	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Les cinq impulsions dans une même piste sont codées avec 20 bits. Les quatre impulsions dans une même piste sont codées avec 16 bits.

Cela donne un total de 72 bits (20+20+16+16) pour le code algébrique.

5.8.1.3 Mode 18,25 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 16 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chaque piste contient quatre impulsions, comme indiqué dans le Tableau 6.

Tableau 6/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 18,25 kbit/s

Piste	Impulsion	Positions
1	i_0, i_4, i_8, i_{12}	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1, i_5, i_9, i_{13}	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6, i_{10}, i_{14}	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7, i_{11}, i_{15}	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Les quatre impulsions dans une même piste sont codées avec 16 bits.

Cela donne un total de 64 bits (16+16+16+16) pour le code algébrique.

5.8.1.4 Mode 15,85 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 12 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chaque piste contient trois impulsions, comme indiqué dans le Tableau 7.

Tableau 7/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 15,85 kbit/s

Piste	Impulsion	Positions
1	i_0, i_4, i_8	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1, i_5, i_9	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6, i_{10}	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7, i_{11}	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Les trois impulsions dans une même piste sont codées avec 13 bits.

Cela donne un total de 52 bits (13+13+13+13) pour le code algébrique.

5.8.1.5 Mode 14,25 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 10 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions d'une sous-trame sont divisées en 4 pistes, où chaque piste contient deux ou trois impulsions, comme indiqué dans le Tableau 8.

Tableau 8/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 14,25 kbit/s

Piste	Impulsion	Positions
1	i_0, i_4, i_8	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1, i_5, i_9	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Chaque paire de positions d'impulsion dans une même piste est codée avec 8 bits (4 bits pour la position de toutes les impulsions), et le signe de la première impulsion dans les pistes est codé avec 1 bit.

Les trois impulsions d'une même piste sont codées avec 13 bits.

Cela donne un total de 44 bits (13+13+9+9) pour le code algébrique.

5.8.1.6 Mode 12,65 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 8 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chaque piste contient deux impulsions, comme indiqué dans le Tableau 9.

Tableau 9/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 12,65 kbit/s

Piste	Impulsion	Positions
1	i_0, i_4	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1, i_5	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2, i_6	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3, i_7	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Chaque paire de positions d'impulsion dans une même piste est codée avec 8 bits (total de 32 bits, 4 bits pour la position de toutes les impulsions), et le signe de la première impulsion dans les pistes est codé avec 1 bit (total de 4 bits). Cela donne un total de 36 bits pour le code algébrique.

5.8.1.7 Mode 8,85 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 4 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 4 pistes, où chaque piste contient une impulsion, comme indiqué dans le Tableau 10.

Tableau 10/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 8,85 kbit/s

Piste	Impulsion	Positions
1	i_0	0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60
2	i_1	1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61
3	i_2	2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54, 58, 62
4	i_3	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43, 47, 51, 55, 59, 63

Chaque position d'impulsion dans une même piste est codée avec 4 bits et le signe de l'impulsion dans les pistes est codé avec 1 bit. Cela donne un total de 20 bits pour le code algébrique.

5.8.1.8 Mode 6,60 kbit/s

Dans ce répertoire, le vecteur d'innovation contient 2 impulsions non nulles. Toutes les impulsions peuvent avoir les amplitudes +1 ou -1. Les 64 positions dans une sous-trame sont divisées en 2 pistes, où chaque piste contient une impulsion, comme indiqué dans le Tableau 11.

Tableau 11/G.722.2 – Positions potentielles d'impulsions individuelles dans le répertoire algébrique, 6,60 kbit/s

Piste	Impulsion	Positions
1	i_0	0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62
2	i_1	1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63

Chaque position d'impulsion dans une même piste est codée avec 5 bits et le signe de l'impulsion dans les pistes est codé avec 1 bit. Cela donne un total de 12 bits pour le code algébrique.

5.8.2 Indexation des impulsions

Dans le paragraphe 5.8.1, le nombre de bits nécessaires pour coder un certain nombre d'impulsions dans une piste était donné. Dans le présent paragraphe, les procédures utilisées pour coder de 1 à 6 impulsions par piste seront décrites. La description sera donnée pour le cas de 4 pistes par sous-trame, avec 16 positions par piste et espacement des impulsions de 4 (ce qui est le cas pour tous les modes sauf le mode 6,6 kbit/s).

Codage de 1 impulsion signée par piste

L'indice de position d'impulsion est codé avec 4 bits et l'indice de signe avec 1 bit. L'indice de position est donné par la position d'impulsion dans la sous-trame divisée par l'espacement des impulsions (division d'entiers). Le reste de la division donne l'indice de piste. Par exemple, une impulsion à la position 31 possède un indice de position de $31/4 = 7$ et appartient à une piste avec indice 3 (4^e piste).

L'indice de signe est ici mis à 0 pour les signes positifs et 1 pour les signes négatifs.

L'indice de l'impulsion signée est donné par

$$I_{1p} = p + s \times 2^M$$

où p est l'indice de position, s est l'indice de signe, et $M = 4$ est le nombre de bits par piste.

Codage de 2 impulsions signées par piste

Dans le cas de deux impulsions par piste avec $K = 2^M$ positions potentielles (ici $M = 4$), chaque impulsion nécessite 1 bit pour le signe et M bits pour la position, ce que donne un total de $2M + 2$ bits. Cependant, une certaine redondance existe en raison de l'insignifiance de l'ordonnancement des impulsions. Par exemple, l'insertion de la première impulsion à la position p et de la deuxième impulsion à la position q équivaut à l'insertion de la première impulsion à la position q et de la deuxième impulsion à la position p . Un seul bit peut être économisé par codage d'un seul signe et déduction du deuxième signe de l'ordonnancement des positions dans l'indice. Ici l'indice est donné par:

$$I_{2p} = p_1 + p_0 \times 2^M + s \times 2^{2M}$$

où s est l'indice de signe de l'impulsion à l'indice de position p_0 . Si les deux signes sont égaux, la plus petite position est mise à p_0 et la plus grande position est mise à p_1 . D'autre part, si les deux signes ne sont pas égaux, la plus grande position est mise à p_0 et la plus petite position est mise à p_1 . Dans le décodeur, le signe de l'impulsion à la position p_0 est immédiatement disponible. Le deuxième signe est déduit de l'ordonnancement des impulsions. Si p_0 est plus grand que p_1 , le signe de l'impulsion à la position p_1 est opposé à celui de la position p_0 . Si ce n'est pas le cas, alors les deux signes sont rendus égaux.

Codage de 3 impulsions signées par piste

Dans le cas de trois impulsions par piste, un raisonnement analogue à celui de deux impulsions peut être utilisé. Pour une piste avec 2^M positions, $3M + 1$ bits sont requis au lieu de $3M + 3$ bits. Une méthode simple d'indexage des impulsions consiste à diviser les positions de piste en deux sections (ou moitiés) et à identifier une section qui contient au moins deux impulsions. Le nombre de positions dans les sections est $K/2 = 2^M/2 = 2^{M-1}$, ce que peut être représenté avec $M-1$ bits. Les deux impulsions dans les sections contenant au moins deux impulsions sont codées avec les procédures pour coder 2 impulsions signées, ce que nécessite $2(M-1)+1$ bits. L'impulsion restante peut être n'importe où dans les pistes (dans l'une ou l'autre section) est codée avec le bit $M + 1$ bits.

Finalement, l'indice de la section qui contient les deux impulsions est codé sur 1 bit. Ainsi le nombre total de bit requis est $2(M-1) + 1 + M + 1 + 1 = 3M + 1$.

Une manière simple de vérifier si deux impulsions sont positionnées dans les mêmes sections est effectuée par vérification de l'égalité ou de l'inégalité du bit de poids fort (MSB, *most significant bit*) de leur indice de position. Noter qu'un MSB de 0 signifie que la position appartient à la moitié inférieure de la piste (0-7) et qu'un bit MSB de 1 signifie qu'il appartient à la moitié supérieure (8-15). Si les deux impulsions appartiennent à la moitié supérieure, il faut les transférer à l'étendue (0-7) avant leur codage au moyen de $2 \times 3 + 1$ bits. Cela peut être effectué par masquage du $M-1$ bit de poids faible (LSB, *least significant bit*) avec un masque composé de $M-1$ uns (ce qui correspond au nombre 7 dans ce cas).

L'indice des 3 impulsions signées est donné par:

$$I_{3p} = I_{2p} + k \times 2^{2M-1} + I_{1p} \times 2^{2M}$$

où I_{2p} est l'indice des deux impulsions dans la même section, où k est l'indice de section (0 ou 1), et où I_{1p} est l'indice de la troisième impulsion dans la piste.

Codage de 4 impulsions signées par piste

Les 4 impulsions signées dans une piste de longueur $K = 2^M$ peuvent être codées au moyen de $4M$ bits. Comme dans le cas de 3 impulsions, les K positions dans les pistes sont divisées en 2 sections (deux moitiés) où chaque section contient $K/2 = 8$ positions. Ici l'on indique les sections comme section A avec les positions de 0 à $K/2-1$ et comme section B avec les positions de $K/2$ à $K-1$. Chaque section peut contenir de 0 à 4 impulsions. Le tableau ci-dessous montre les 5 cas représentant le nombre possible d'impulsions dans chaque section:

Cas	Impulsions à la section A	Impulsions à la section B	Bits requis
0	0	4	$4M-3$
1	1	3	$4M-2$
2	2	2	$4M-2$
3	3	1	$4M-2$
4	4	0	$4M-3$

Dans le cas 0 ou 4, les 4 impulsions dans une section de longueur $K/2 = 2^{M-1}$ peuvent être codées au moyen de $4(M-1)+1 = 4M-3$ bits (cela sera expliqué ultérieurement).

Dans le cas 1 ou 3, l'unique impulsion dans une section de longueur $K/2 = 2^{M-1}$ peut être codée avec $M-1+1 = M$ bits et les 3 impulsions dans les autres sections peuvent être codées avec $3(M-1)+1 = 3M-2$ bits. Cela donne un total de $M+3M-2 = 4M-2$ bits.

Dans le cas 2, les impulsions dans une section de longueur $K/2 = 2^{M-1}$ peuvent être codées avec $2(M-1)+1 = 2M-1$ bit. Ainsi pour les deux sections, $2(2M-1) = 4M-2$ bits sont requis.

Or l'indice de cas peut être codé avec 2 bits (4 cas possibles) dans l'hypothèse où les cas 0 et 4 sont combinés. Pour le cas 1, 2, ou 3, le nombre de bits requis est donc de $4M-2$. Cela donne un total de $4M-2 + 2 = 4M$ bits. Pour le cas 0 ou 4, un seul bit est requis afin d'identifier chaque cas, et $4M-3$ bits sont requis pour coder les 4 impulsions dans les sections. En ajoutant les 2 bits requis pour le cas général, cela donne un total de $1+4M-3+2 = 4M$ bits.

L'indice des 4 impulsions signées est donné par:

$$I_{4p} = I_{AB} + k \times 2^{4M-2}$$

où k est l'indice de cas (2 bits), et I_{AB} est l'indice des impulsions dans les deux sections pour chaque cas individuel.

Pour les cas 0 et 1, I_{AB} est donné par:

$$I_{AB_{0,4}} = I_{4p_section} + j \times 2^{4M-3}$$

où j est un fanion identificateur à 1 bit de la section avec 4 impulsions et $I_{4p_section}$ est l'indice des 4 impulsions dans cette section (que nécessite $4M-3$ bits).

Pour le cas 1, I_{AB} est donné par:

$$I_{AB_1} = I_{3p_B} + I_{1p_A} \times 2^{3(M-1)+1}$$

où I_{3p_B} est l'indice des 3 impulsions à la section B ($3(M-1)+1$ bits) et I_{1p_A} est l'indice de l'impulsion à la section A ($(M-1)+1$ bits).

Pour le cas 2, I_{AB} est donné par:

$$I_{AB_2} = I_{2p_B} + I_{2p_A} \times 2^{2(M-1)+1}$$

où I_{2p_B} est l'indice des 2 impulsions à la section B ($2(M-1)+1$ bits) et I_{2p_A} est l'indice des 2 impulsions à la section A ($2(M-1)+1$ bits).

Finalement, pour le cas 3, I_{AB} est donné par:

$$I_{AB_3} = I_{1p_B} + I_{3p_A} \times 2^M$$

où I_{1p_B} est l'indice de l'impulsion à la section B ($(M-1)+1$ bits) et I_{3p_A} est l'indice des 3 impulsions à la section A ($3(M-1)+1$ bits).

Pour les cas 0 et 4, il a été mentionné que les 4 impulsions dans une section sont codées au moyen de $4(M-1)+1$ bits. Cela est effectué par division ultérieure de la section en 2 sous-sections de longueur $K/4 = 2^{M-2}$ ($=4$ dans ce cas); par identification d'une sous-section qui contient au moins 2 impulsions; par codage des 2 impulsions dans cette sous-section au moyen de $2(M-2)+1 = 2M-3$ bits; par codage de l'indice de la sous-section qui contient au moins 2 impulsions, au moyen de 1 bit; et par codage des 2 impulsions restantes, dans l'hypothèse qu'elles peuvent être n'importe où dans les sections, au moyen de $2(M-1)+1 = 2M-1$ bits. Cela donne un total de $(2M-3)+(1)+(2M-1) = 4M-3$ bits.

Codage de 5 impulsions signées par piste

Les 5 impulsions signées dans une piste de longueur $K = 2^M$ peuvent être codées au moyen de $5M$ bits. Comme dans le cas de 4 impulsions, les K positions dans les pistes sont divisées en 2 sections A et B. Chaque section peut contenir de 0 à 5 impulsions. Une méthode simple de codage de 5 impulsions consiste à identifier une section qui contient au moins 3 impulsions puis à coder 3 impulsions dans cette section au moyen de $3(M-1)+1 = 3M-2$ bits, puis à coder les 2 impulsions restantes dans la piste entière au moyen de $2M+1$ bits. Cela donne $5M-1$ bits. Un bit supplémentaire est nécessaire pour identifier la section qui contient au moins 3 impulsions. Ainsi un total de $5M$ bits est requis afin de coder 5 impulsions signées.

L'indice des 5 impulsions signées est donné par:

$$I_{5p} = I_{2p} + I_{3p} \times 2^{2M} + k \times 2^{5M-1}$$

où k est l'indice de la section qui contient au moins 3 impulsions, I_{3p} est l'indice des 3 impulsions dans cette section ($3(M-1)+1$ bits), et I_{2p} est l'indice des 2 impulsions restant dans les pistes ($2M+1$ bits).

Codage de 6 impulsions signées par piste

Les 6 impulsions signées dans une piste de longueur $K = 2^M$ sont codées au moyen de $6M-2$ bits. Comme dans le cas de 5 impulsions, les K positions dans les pistes sont divisées en 2 sections A et B. Chaque section peut contenir de 0 à 6 impulsions. Le tableau ci-dessous montre les 7 cas représentant le nombre possible d'impulsions dans chaque section:

Cas	Impulsions à la section A	Impulsions à la section B	Bits requis
0	0	6	$6M-5$
1	1	5	$6M-5$
2	2	4	$6M-5$
3	3	3	$6M-4$
4	4	2	$6M-5$
5	5	1	$6M-5$
6	6	0	$6M-5$

Noter que les cas 0 et 6 sont similaires sauf que les 6 impulsions sont dans des sections différentes. De la même façon, les cas 1 et 5 ainsi que les cas 2 et 4 diffèrent seulement dans les sections qui contiennent plus d'impulsions. Donc ces cas peuvent être couplés et un bit supplémentaire peut être attribué afin d'identifier la section qui contient plus d'impulsions. Etant donné que ces cas nécessitent initialement $6M-5$ bits, les cas couplés nécessitent $6M-4$ bits compte tenu du bit de section. Ainsi, nous avons maintenant 4 états de cas couplés, c'est-à-dire (0,6), (1,5), (2,4), et (3), avec 2 bits supplémentaires requis pour indiquer l'état. Cela donne un total de $6M-4+2 = 6M-2$ bits pour les 6 impulsions signées.

Dans les cas 0 et 6, 1 bit est nécessaire pour identifier la section qui contient 6 impulsions. Cinq impulsions dans cette section sont codées au moyen de $5(M-1)$ bits (étant donné que les impulsions sont limitées à cette section), et l'impulsion restante est codée au moyen de $(M-1)+1$ bits. Ainsi un total de $1+5(M-1)+M = 6M-4$ bits est requis pour ce cas couplé. Deux bits supplémentaires sont requis afin de coder l'état du cas couplé, ce qui donne un total de $6M-2$ bits. Pour ce cas couplé, l'indice des 6 impulsions est donné par:

$$I_{6p} = I_{1p} + I_{5p} \times 2^M + j \times 2^{6M-5} + k \times 2^{6M-4}$$

où k est l'indice du cas couplé (2 bits), j est l'indice de la section contenant 6 impulsions (1 bit), I_{5p} est l'indice de 5 impulsions dans cette section ($5(M-1)$ bits), et I_{1p} est l'indice de l'impulsion restante dans cette section ($(M-1)+1$ bits).

Dans les cas 1 et 5, 1 bit est nécessaire pour identifier la section qui contient 5 impulsions. Les 5 impulsions dans cette section sont codées au moyen de $5(M-1)$ bits et l'impulsion dans les autres sections est codée au moyen de $(M-1)+1$ bits. Pour ce cas couplé, l'indice des 6 impulsions est donné par:

$$I_{6p} = I_{1p} + I_{5p} \times 2^M + j \times 2^{6M-5} + k \times 2^{6M-4}$$

où k est l'indice du cas couplé (2 bits), j est l'indice de la section contenant 5 impulsions (1 bit), I_{5p} est l'indice des 5 impulsions dans cette section ($5(M-1)$ bits), et I_{1p} est l'indice de l'impulsion dans les autres sections ($(M-1)+1$ bits).

Dans le cas 2 ou 4, 1 bit est nécessaire pour identifier la section qui contient 4 impulsions. Les 4 impulsions dans cette section sont codées au moyen de $4(M-1)$ bits et les 2 impulsions dans les autres sections sont codées au moyen de $2(M-1)+1$ bits. Pour ce cas couplé, l'indice des 6 impulsions est donné par:

$$I_{6p} = I_{2p} + I_{4p} \times 2^{2(M-1)+1} + j \times 2^{6M-5} + k \times 2^{6M-4}$$

où k est l'indice du cas couplé (2 bits), j est l'indice de la section contenant 4 impulsions (1 bit), I_{4p} est l'indice de 4 impulsions dans cette section ($4(M-1)$ bits), et I_{2p} est l'indice des 2 impulsions dans les autres sections ($2(M-1)+1$ bits).

Dans le cas 3, les 3 impulsions dans chaque section sont codées au moyen de $3(M-1)+1$ bits dans chaque section. Pour ce cas, l'indice des 6 impulsions est donné par:

$$I_{6p} = I_{3pB} + I_{3pA} \times 2^{3(M-1)+1} + k \times 2^{6M-4}$$

où k est l'indice du cas couplé (2 bits), I_{3pB} est l'indice de 3 impulsions section B ($3(M-1)+1$ bits), et I_{3pA} est l'indice des 3 impulsions à la section A ($3(M-1)+1$ bits).

5.8.3 Exploration du répertoire

Le répertoire algébrique est exploré par minimisation de l'erreur quadratique moyenne entre le signal vocal d'entrée pondéré et le signal vocal de synthèse pondéré. Le signal cible utilisé dans la recherche tonale en boucle fermée est mis à jour par soustraction de la contribution du répertoire adaptatif. Soit

$$x_2(n) = x(n) - g_p y(n), \quad n = 0, \dots, 63 \quad (40)$$

où $y(n) = v(n) * h(n)$ est le vecteur filtré du répertoire adaptatif et g_p est le gain non quantifié du répertoire adaptatif.

La matrice \mathbf{H} est définie comme étant la partie triangulaire inférieure de la matrice convolutionnelle de Töplitz avec diagonale $h(0)$ et diagonales inférieures $h(1), \dots, h(63)$, et $\mathbf{d} = \mathbf{H}^t \mathbf{x}_2$ est la corrélation entre le signal cible $x_2(n)$ et la réponse impulsionnelle $h(n)$ (également appelé "vecteur cible filtré en inverse"), et $\Phi = \mathbf{H}^t \mathbf{H}$ est la matrice de corrélations de $h(n)$.

Les éléments du vecteur \mathbf{d} sont calculés par:

$$d(n) = \sum_{i=n}^{63} x_2(i) h(i-n), \quad n = 0, \dots, 63 \quad (41)$$

et les éléments de la matrice symétrique Φ sont calculés par:

$$\phi(i, j) = \sum_{n=j}^{63} h(n-i) h(n-j), \quad i = 0, \dots, 63 \quad j = i, \dots, 63 \quad (42)$$

Si \mathbf{c}_k est le vecteur de code algébrique à l'indice k , le répertoire algébrique est exploré par maximisation du critère de recherche:

$$Q_k = \frac{(\mathbf{x}_2^t \mathbf{H} \mathbf{c}_k)^2}{\mathbf{c}_k^t \mathbf{H}^t \mathbf{H} \mathbf{c}_k} = \frac{(\mathbf{d}^t \mathbf{c}_k)^2}{\mathbf{c}_k^t \Phi \mathbf{c}_k} = \frac{(R_k)^2}{E_k} \quad (43)$$

Le vecteur \mathbf{d} et la matrice Φ sont habituellement calculées avant à la recherche dans le répertoire.

La structure algébrique du répertoire autorise des procédures de recherche très rapides étant donné que le vecteur d'innovation \mathbf{c}_k contient seulement quelques impulsions non nulles. La corrélation dans le numérateur de l'équation (43) est donné par:

$$C = \sum_{i=0}^{N_p-1} a_i d(m_i) \quad (44)$$

où m_i est la position de la $i^{\text{ème}}$ impulsion, a_i est son amplitude, et N_p est le nombre d'impulsions. L'énergie dans le dénominateur de l'équation (43) est donnée par:

$$E = \sum_{i=0}^{N_p-1} \phi(m_i, m_i) + 2 \sum_{i=0}^{N_p-2} \sum_{j=i+1}^{N_p-1} a_i a_j \phi(m_i, m_j) \quad (45)$$

Afin de simplifier la procédure de recherche, les amplitudes d'impulsion sont prédéterminées sur la base d'un certain signal de référence $b(n)$. Dans cette méthode d'amplitude d'impulsion sélectionnée par le signal, le signe d'une impulsion à la position i est rendu égal au signe du signal de référence à cette position. Ici, le signal de référence $b(n)$ est donné par:

$$b(n) = \sqrt{\frac{E_d}{E_r}} r_{LTP}(n) + \alpha d(n) \quad (46)$$

où $E_d = \mathbf{d}^t \mathbf{d}$ est l'énergie du signal $d(n)$ et $E_r = \mathbf{r}_{LTP}^t \mathbf{r}_{LTP}$ est l'énergie du signal $r_{LTP}(n)$ qui est le signal résiduel après prédiction à long terme. Le facteur de normalisation α commande le degré de dépendance du signal de référence de $d(n)$, et il est diminué lorsque le débit est augmenté. Ici $\alpha = 2$ pour les modes 6,6 et 8,85; $\alpha = 1$ pour les modes 12,65, 14,25, et 15,85; $\alpha = 0,8$ pour 18,25; $\alpha = 0,75$ pour 19,85; et $\alpha = 0,5$ pour 23,05 et 23,85.

Afin de simplifier la recherche, le signal $d(n)$ et la matrice Φ sont modifiés de façon à incorporer les signes présélectionnés. Soit $s_b(n)$ le vecteur contenant les signes de $b(n)$. Le signal modifié $d'(n)$ est donné par:

$$d'(n) = s_b(n) d(n) \quad n = 0, \dots, N-1$$

et la matrice d'autocorrélation Φ' est donnée par:

$$\phi'(i, j) = s_b(i) s_b(j) \phi(i, j), \quad i = 0, \dots, N-1; \quad j = i, \dots, N-1$$

La corrélation au numérateur du critère de recherche Q_k est donc donnée par:

$$R = \sum_{i=0}^{N_p-1} d'(i)$$

et l'énergie au dénominateur du critère de recherche Q_k est donnée par:

$$E = \sum_{i=0}^{N_p-1} \phi'(m_i, m_i) + 2 \sum_{i=0}^{N_p-2} \sum_{j=i+1}^{N_p-1} \phi'(m_i, m_j)$$

Le but de la recherche consiste donc à déterminer le vecteur de code avec le meilleur ensemble de N_p positions d'impulsion dans l'hypothèse où les amplitudes des impulsions ont été sélectionnées comme décrit ci-dessus. Le basic critère de sélection de base est la maximisation du rapport Q_k susmentionné.

Afin de réduire la complexité de la recherche, une procédure rapide de recherche dite *procédure de recherche arborescente en profondeur d'abord* est utilisée, dans laquelle les positions d'impulsion sont déterminées N_m impulsions à la fois. Plus précisément, les N_p impulsions

disponibles sont réparties en M sous-ensembles non vides de N_m impulsions chacun, tels que $N_1+N_2...+N_m...+N_M = N_p$. Un choix particulier de positions pour les $J = N_1+N_2...+N_{m-1}$ premières impulsions considérées est appelé *trajet de niveau- m* ou *trajet de longueur J* . Le critère de base pour un trajet de J positions d'impulsion est le rapport $Q_k(J)$ lorsque seules les J impulsions correspondantes sont considérées.

La recherche commence par le sous-ensemble #1 et continue par les sous-ensembles suivants, conformément à une structure arborescente dans laquelle le sous-ensemble m est exploré au $m^{\text{ième}}$ niveau de l'arbre. Le but de la recherche au niveau 1 consiste à considérer les N_1 impulsions du sous-ensemble #1 et leurs positions valides afin de déterminer un ou plusieurs trajets candidats de longueur N_1 qui sont les nœuds d'arbre au niveau 1. Le trajet à chaque nœud terminal de niveau $m-1$ est étendu à la longueur $N_1+N_2...+N_m$ au niveau m par considération de N_m impulsions nouvelles avec leurs positions valides. Un ou plusieurs trajets étendus candidats sont déterminés afin de constituer des nœuds de niveau- m . Le meilleur vecteur de code correspond au trajet de longueur N_p qui maximise le critère $Q_k(N_p)$ en ce qui concerne tous les nœuds de niveau- M .

Une forme spéciale de la procédure de recherche arborescente en profondeur d'abord est utilisée ici, dans laquelle deux impulsions sont recherchées à la fois, soit, $N_m=2$. Ces 2 impulsions appartiennent à deux pistes consécutives. Par ailleurs, au lieu de supposer que la matrice Φ est précalculée et mémorisée, ce qui nécessite une mémoire de $N \times N$ mots ($64 \times 64 = 4k$ mots), une méthode utilisant efficacement la mémoire est utilisée afin de réduire la mémoire requise. Dans cette méthode, la procédure de recherche est effectuée de telle façon que seule une partie des éléments requis de la matrice de corrélation est précalculée et mémorisée. Cette partie correspond à la corrélation de la réponse impulsionnelle correspondant à la position potentielle d'impulsions dans des pistes consécutives, ainsi qu'aux corrélations correspondant à $\phi(j,j)$, $j = 0, \dots, N-1$ (qui sont les éléments de la diagonale principale de matrice Φ).

Afin de réduire la complexité, tout en contrôlant d'éventuelles combinaisons de deux impulsions, un nombre limité de positions potentielles de la première impulsion est essayé. Par ailleurs, s'il y a un grand nombre d'impulsions, certaines impulsions dans les niveaux supérieurs de la recherche arborescente sont fixes. Afin de supposer de façon intelligente quelles sont les positions potentielles d'impulsion qui sont considérées pour la première impulsion ou afin de fixer certaines positions d'impulsion, un "vecteur d'estimation de la probabilité de position d'impulsion" \mathbf{b} est utilisé, qui est fondé sur des signaux associés à la parole. Le $p^{\text{ième}}$ composant $b(p)$ de ce vecteur estimé \mathbf{b} caractérise la probabilité d'une impulsion occupant la position p ($p = 0, 1, \dots, N-1$) dans le meilleur vecteur de code recherché. Ici le vecteur estimé \mathbf{b} est celui qui est utilisé pour présélectionner les amplitudes et qui est donné dans l'équation (46).

Les procédures de recherche pour tous les modes de débit sont similaires. Deux impulsions sont recherchées à la fois et ces deux impulsions correspondent toujours à des pistes consécutives. C'est-à-dire que les deux impulsions recherchées sont dans les pistes T_0-T_1 , T_1-T_2 , T_2-T_3 , ou T_3-T_0 .

Avant de rechercher les positions, le signe d'une impulsion à une position possible n est considéré comme étant le signe de $b(n)$ à cette position. Ensuite le signal modifié $d'(n)$ est calculé comme décrit ci-dessus par y insertion des signes prédéterminés.

Pour les deux premières impulsions (1^{er} niveau de l'arborescence), la corrélation au numérateur du critère de recherche est donnée par:

$$R = d'(m_0) + d'(m_1)$$

et l'énergie au dénominateur du critère de recherche Q_k est donné par:

$$E = \phi'(m_0, m_0) + \phi'(m_1, m_1) + 2\phi'(m_0, m_1)$$

où les corrélations $\phi'(m_i, m_j)$ ont été modifiées de façon à inclure les signes présélectionnés aux positions m_i et m_j .

Pour les niveaux subséquents, le numérateur et le dénominateur sont mis à jour par adjonction de la contribution de deux nouvelles impulsions. Dans l'hypothèse où deux nouvelles impulsions sont recherchées à un certain niveau d'arborescence avec les positions m_k et m_{k+1} de deux pistes consécutives, la valeur mise à jour de R est donnée par:

$$R = R + d'(m_k) + d'(m_{k+1}) \quad (47)$$

et l'énergie mise à jour est donnée par:

$$E = E + \phi'(m_k, m_k) + \phi'(m_{k+1}, m_{k+1}) + 2\phi'(m_k, m_{k+1}) + 2R_{hv}(m_k) + 2R_{hv}(m_{k+1}) \quad (48)$$

où $R_{hv}(m)$ est la corrélation entre la réponse impulsionnelle $h(n)$ et un vecteur $v_h(n)$ contenant l'adjonction de versions retardées de réponse impulsionnelle aux positions déjà déterminées. Soit,

$$v_h(n) = \sum_{i=0}^{k-1} h(n - m_i)$$

et

$$R_{hv}(m) = \sum_{n=m}^{N-1} h(n)v_h(n-m)$$

A chaque niveau d'arborescence, les valeurs de $R_{hv}(m)$ sont calculées en ligne pour toutes les positions possibles dans chacune des 2 pistes essayées. Il ressort de l'équation (48) que seules les corrélations $\phi'(m_k, m_{k+1})$ correspondant aux positions d'impulsion dans deux pistes consécutives ont besoin d'être mémorisées ($4 \times 16 \times 16$ mots), de même que les corrélations $\phi'(m_k, m_k)$ correspondant à la diagonale de la matrice Φ (64 mots). La mémoire requise dans la présente structure algébrique est donc de 1088 mots au lieu de $64 \times 64 = 4096$ mots.

Les procédures de recherche aux différents modes de débit sont similaires. La différence réside dans le nombre d'impulsions et, en conséquence, dans le nombre de niveaux lors de la recherche arborescente. Afin de conserver une complexité comparable de recherche entre les différents répertoires, le nombre de positions essayées est gardé similaire.

La recherche dans le mode 12,65 kbit/s sera décrite à titre d'exemple. Dans ce mode, 2 impulsions sont placées dans chaque piste, ce qui donne un total de 8 impulsions par sous-trame de longueur 64. Deux impulsions sont explorées à la fois, et ces deux impulsions correspondent toujours à des pistes consécutives. C'est-à-dire que les deux impulsions recherchées sont dans les pistes T_0-T_1 , T_1-T_2 , T_2-T_3 , ou T_3-T_0 . L'arbre possède 4 niveaux dans ce cas. Au premier niveau, l'impulsion P_0 est attribuée à la piste T_0 et l'impulsion P_1 à la piste T_1 . A ce niveau, aucune recherche n'est effectuée et les deux positions d'impulsion sont mises au maximum de $b(n)$ dans chaque piste. Au deuxième niveau, l'impulsion P_2 est attribuée à la piste T_2 et l'impulsion P_3 à la piste T_3 . Quatre positions pour l'impulsion P_2 sont essayées en fonction des 16 positions de l'impulsion P_3 . Les 4 positions essayées de P_2 sont déterminées sur la base des maxima de $b(n)$ dans les pistes. Au troisième niveau, l'impulsion P_4 est attribuée à la piste T_1 et l'impulsion P_5 à la piste T_2 . Huit positions pour l'impulsion P_4 sont essayées en fonction des 16 positions de l'impulsion P_5 . Comme au niveau de recherche précédent, les 8 positions essayées de P_4 sont déterminées sur la base des maxima de $b(n)$ dans la piste. Au quatrième niveau, l'impulsion P_6 est attribuée à la piste T_3 et l'impulsion P_7 à la piste T_0 . Huit positions pour l'impulsion P_6 sont essayées en fonction des 16 positions de l'impulsion P_7 . Ainsi le nombre total de combinaisons essayées est $4 \times 16 + 8 \times 16 + 8 \times 16 = 320$. L'ensemble du processus est répété 4 fois (4 itérations) par attribution des impulsions à différentes pistes. Par exemple, dans la 2^e itération, les impulsions P_0 à P_7 sont attribuées aux pistes $T_1, T_2, T_3, T_0, T_2, T_3, T_0$, et T_1 , respectivement. Ainsi le nombre total combinaisons de positions essayées est de $4 \times 320 = 1280$.

A titre d'autre exemple de recherche, dans le mode 15,85 kbit/s, 3 impulsions sont placées dans chaque piste, ce qui donne un total de 12 impulsions. Il y a 6 niveaux dans la recherche arborescente, par laquelle deux impulsions sont recherchées dans chaque niveau. Dans les deux premiers niveaux, 4 impulsions sont mises aux maxima de $b(n)$. Dans les 4 niveaux subséquents, les nombres de combinaisons essayées sont respectivement de 4×16 , 6×16 , 8×16 et de 8×16 . 4 itérations sont utilisées, ce qui donne un total de $4 \times 26 \times 16 = 1664$ combinaisons.

5.9 Quantification des gains de répertoire adaptatif et de répertoire fixe

Le gain du répertoire adaptatif (gain tonal) et le gain de répertoire fixe (algébrique) sont quantifiés vectoriellement au moyen d'un répertoire de 6 bits pour les modes 8,85 et 6,60 kbit/s et au moyen d'un répertoire de 7 bits pour tous les autres modes.

La quantification du gain du répertoire fixe est effectuée au moyen de la prédiction à moyenne mobile (MA) avec coefficients fixes. La prédiction à moyenne mobile (MA) du 4^e ordre est effectuée sur l'énergie d'innovation comme suit. Soit $E(n)$ l'énergie d'innovation après suppression de la moyenne (en dB) à la sous-trame n , donnée par:

$$E(n) = 10 \log \left(\frac{1}{N} g_c^2 \sum_{i=0}^{N-1} c^2(i) \right) - \bar{E} \quad (49)$$

où $N = 64$ est la longueur de sous-trame, $c(i)$ est l'excitation par le répertoire fixe, et $\bar{E} = 30$ dB est la moyenne de l'énergie d'innovation. L'énergie prédite est donnée par:

$$\tilde{E}(n) = \sum_{i=1}^4 b_i \hat{R}(n-i) \quad (50)$$

où $[b_1 \ b_2 \ b_3 \ b_4] = [0,5, 0,4, 0,3, 0,2]$ sont les coefficients de prédiction MA, et où $\hat{R}(k)$ est l'erreur de prédiction de l'énergie quantifiée à la sous-trame k . L'énergie prédite est utilisée pour calculer un gain prédit du répertoire fixe g'_c comme dans l'équation (49) (par remplacement de $E(n)$ par $\tilde{E}(n)$ et de g_c par g'_c), ce qui est effectué comme suit. L'énergie d'innovation moyenne est d'abord trouvée par:

$$E_i = 10 \log \left(\frac{1}{N} \sum_{i=0}^{N-1} c^2(i) \right) \quad (51)$$

puis le gain prédit g'_c est trouvé par:

$$g'_c = 10^{0,05(\tilde{E}(n) + \bar{E} - E_i)} \quad (52)$$

Un facteur de correction entre le gain g_c et le gain estimé g'_c est donné par:

$$\gamma = g_c / g'_c \quad (53)$$

Noter que l'erreur de prédiction est donnée par:

$$R(n) = E(n) - \tilde{E}(n) = 20 \log(\gamma) \quad (54)$$

Le gain tonal, g_p , et le facteur de correction γ sont quantifiés vectoriellement de concert au moyen d'un répertoire de 6 bits pour les modes 8,85 et 6,60 kbit/s, et d'un répertoire de 7 bits pour les autres modes. La recherche de gain dans le répertoire est effectuée par minimisation de l'erreur quadratique pondérée entre signal de parole original et signal de parole reconstruit, ce qui est donné par:

$$E = x^t x + g_p^2 y^t y + g_c^2 z^t z - 2g_p x^t y - 2g_c x^t z + 2g_p g_c y^t z \quad (55)$$

où la variable x est le vecteur cible, y est le vecteur filtré du répertoire adaptatif, et z est le vecteur filtré du répertoire fixe. (Chaque vecteur de gain possède également dans le répertoire un élément représentant l'erreur de prédiction sur l'énergie quantifiée.) L'erreur de prédiction sur l'énergie quantifiée, associée aux gains choisis, est utilisée pour mettre à jour $\hat{R}(n)$. Dans la recherche, seuls les 64 vecteurs de code qui sont les plus proches du gain tonal non quantifié, g_p , sont pris en compte.

5.10 Mise à jour de la mémoire

Une mise à jour des états du filtre de synthèse et du filtre de pondération est requise afin de calculer le signal cible dans la sous-trame suivante.

Après que les deux gains ont été quantifiés, le signal d'excitation, $u(n)$, est trouvé dans la sous-trame présente par

$$u(n) = \hat{g}_p v(n) + \hat{g}_c c(n), \quad n = 0, \dots, 63 \quad (56)$$

où \hat{g}_p et \hat{g}_c sont respectivement les gains quantifiés des répertoires adaptatif et fixe, où $v_i(n)$ est le vecteur du répertoire adaptatif (excitation antérieure interpolée), et $c(n)$ est le vecteur du répertoire fixe (code algébrique y compris l'affinage du délai tonal). Les états des filtres peuvent être mis à jour par filtrage du signal $r(n) - u(n)$ (différence entre signal résiduel et excitation) dans les filtres $1/\hat{A}(z)$ et $A(z/\gamma_1)H_{\text{de-emph}}(z)$ pour les 64 sous-frames échantillons et par mémorisation des états des filtres, ce qui nécessitera 3 filtrages. Une méthode plus simple, qui ne nécessite qu'un seul filtrage, est la suivante. Le signal de synthèse local, $\hat{s}(n)$, est calculé par filtrage du signal d'excitation dans $1/\hat{A}(z)$. La sortie du filtre en fonction de l'entrée $r(n) - u(n)$ est équivalent à $e(n) = s(n) - \hat{s}(n)$. Les états du filtre de synthèse $1/\hat{A}(z)$ sont donc donnés par $e(n)$, $n = 48, \dots, 63$. La mise à jour des états du filtre $A(z/\gamma_1)H_{\text{de-emph}}(z)$ peut être effectuée par filtrage du signal d'erreur $e(n)$ dans ce filtre afin de trouver l'erreur pondérée par la perception $e_w(n)$. Cependant, le signal $e_w(n)$ peut être aussi bien trouvé par:

$$e_w(n) = x(n) - \hat{g}_p y(n) = \hat{g}_c z(n) \quad (57)$$

Etant donné que les signaux $x(n)$, $y(n)$, et $z(n)$ sont disponibles, les états du filtre de pondération sont mis à jour par calcul de $e_w(n)$ comme dans l'équation (54) pour $n = 48, \dots, 63$, ce qui économise deux filtrages.

5.11 Production du gain dans la bande des hautes fréquences

Afin de calculer le gain dans la bande des hautes fréquences pour le mode 23,85 kbit/s mode, le signal vocal d'entrée de 16 kHz est filtré dans un filtre FIR passe-bande $H_{HB}(z)$ que possède une bande passante de 6,4 à 7 kHz. Le gain dans la bande des hautes fréquences g_{HB} est obtenu par

$$g_{HB} = \frac{\sum_{i=0}^{63} (s_{HB}(i))^2}{\sum_{i=0}^{63} (s_{HB2}(i))^2} \quad (58)$$

où $s_{HB}(i)$ est le signal vocal d'entrée après filtrage passe-bande et $s_{HB2}(i)$ est la synthèse vocale en bande haute obtenue de l'excitation en bande haute $u_{HB2}(i)$ après passage dans le filtre de synthèse en bande haute $A_{HB}(z)$ décrit au § 6.3.2.2.

6 Description fonctionnelle du décodeur

La fonction du décodeur se compose du décodage des paramètres transmis (paramètres LP, vecteur du répertoire adaptatif, gain du répertoire adaptatif, vecteur du répertoire fixe, gain du répertoire fixe et gain en bande haute) et de l'exécution d'une synthèse afin d'obtenir le signal vocal reconstruit, lequel subit ensuite un post-traitement et un suréchantillonnage (avec renormalisation multiplicatrice). Finalement le signal en bande haute est produit dans la bande de fréquence de 6 à 7 kHz. Le flux de signaux dans le décodeur est présenté dans la Figure 3.

6.1 Décodage et synthèse de parole

Le processus de décodage est effectué dans l'ordre suivant:

décodage des paramètres de filtre LP: les indices reçus de quantification de paires ISP sont utilisés pour reconstruire le vecteur de paire ISP quantifiée. L'interpolation décrite au § 5.2.6 est effectuée afin d'obtenir 4 vecteurs ISP interpolés (correspondant à 4 sous-trames). Pour chaque sous-trame, le vecteur ISP interpolé est converti dans le domaine des coefficients de filtre LP, a_k , qui est utilisé pour synthétiser le signal vocal reconstruit dans la sous-trame.

Les échelons suivants sont répétés pour chaque sous-trame:

- 1) **décodage du vecteur du répertoire adaptatif:** l'indice tonal reçu (du répertoire adaptatif) est utilisé afin de trouver les parties entière et fractionnaire du délai tonal. Le vecteur du répertoire adaptatif $v(n)$ est trouvé par interpolation de l'excitation antérieure $u(n)$ (au délai tonal) au moyen du filtre FIR décrit au § 5.6. L'indice de filtre adaptatif reçu est utilisé afin de trouver si le répertoire adaptatif filtré est $v_1(n) = v(n)$ or $v_2(n) = 0,18v(n) + 0,64v(n-1) + 0,18v(n-2)$;
- 2) **décodage du vecteur innovatif:** l'indice reçu du répertoire algébrique sert à extraire les positions et amplitudes (signes) de l'excitation impulsions et afin de trouver le vecteur de code algébrique $c(n)$. Si la partie entière du délai tonal est inférieure à la longueur de sous-trame 64, la procédure d'affinage du délai tonal est appliquée, ce qui revient à modifier $c(n)$ par filtrage dans le préfiltre adaptatif $F(z)$, lequel se compose de deux parties: une partie d'amélioration de la périodicité $1/(1-0,85z^{-T})$ et une partie d'écart de niveau à court terme $(1 - \beta_1 z^{-1})$, où T est la partie entière du délai tonal et où $\beta_1(n)$ est associé à la verbalisation de la précédente sous-trame et est limité par $[0,0, 0,5]$;

- 3) **décodage du gain de répertoire adaptatif et de répertoire d'innovation:** l'indice reçu donne le facteur de correction $\hat{\gamma}$ du gain du répertoire fixe. Le gain estimé du répertoire fixe g'_c est trouvé comme décrit au § 5.8. L'énergie prédite pour toutes les sous-frames n est trouvée d'abord par

$$\tilde{E}(n) = \sum_{i=1}^4 b_i \hat{R}(n-i) \quad (59)$$

puis l'énergie moyenne d'innovation est trouvée par:

$$E_i = 10 \log \left(\frac{1}{N} \sum_{i=0}^{N-1} c^2(i) \right) \quad (60)$$

Le gain prédit g'_c est trouvé par:

$$g'_c = 10^{0.05(\tilde{E}(n) + \bar{E} - E_i)} \quad (61)$$

Le gain quantifié du répertoire fixe est donné par:

$$\hat{g}_c = \hat{\gamma} g'_c \quad (62)$$

- 4) **calcul du signal vocal reconstruit:** les échelons suivants sont pour $n = 0, \dots, 63$. L'excitation totale est construite par:

$$u(n) = \hat{g}_p v(n) + \hat{g}_c c(n) \quad (63)$$

Avant la synthèse vocale, un post-traitement des éléments d'excitation est effectué;

- 5) **traitement anti-dispersion (modes 6,60 et 8,85 kbit/s):** une procédure adaptative de post-traitement antidispersion est appliquée au vecteur du répertoire fixe $c(n)$ afin de réduire les artéfacts perceptifs provenant de la dispersion du vecteur du répertoire algébrique fixe avec seulement quelques échantillons non nuls par sous-trame. Le traitement anti-dispersion se compose d'une convolution circulaire du vecteur du répertoire fixe avec une réponse impulsionnelle. Trois réponses impulsionnelles prémémorisées sont utilisées et un nombre $impNr = 0,1,2$ est choisi afin d'en sélectionner une. Une valeur de 2 correspond à aucune modification, une valeur de 1 correspond à une modification moyenne, tandis qu'une valeur de 0 correspond à une modification forte. La sélection de la réponse impulsionnelle est effectuée dynamiquement à partir des gains de répertoire adaptatif et de répertoire fixe. La procédure suivante est employée:

Si $\hat{g}_p < 0,6$ alors

$$impNr = 0;$$

même si $\hat{g}_p < 0,9$ alors

$$impNr = 1;$$

donc

$$impNr = 2;$$

détecter un nouvel événement par comparaison du gain du répertoire fixe avec le précédent gain du répertoire fixe. Si la valeur actuelle est supérieure au triple de la précédente valeur, un nouvel événement est détecté.

S'il n'y a aucun nouvel événement et si $impNr = 0$, la valeur médiane filtrée du gain actuel et des 4 gains précédents du répertoire adaptatif est calculée. Si cette valeur est inférieure à 0,6, $impNr = 0$.

S'il n'y a aucun nouvel événement, la valeur $impNr$ est limitée à augmenter d'un échelon par rapport à la sous-trame précédente.

Si un nouvel événement est déclaré, la valeur $impNr$ est augmentée de 1 si elle est inférieure à 2.

Si le mode est 8,85 kbit/s la valeur $impNr$ est augmentée de un;

- 6) **renforceur d'excitation en terme de bruit**: une technique de lissage non linéaire du gain est appliquée au gain du répertoire fixe \hat{g}_c afin de renforcer l'excitation en terme de bruit. Sur la base de la stabilité et de la verbalisation du segment vocal, le gain du répertoire fixe est lissé afin de réduire la fluctuation de l'énergie de l'excitation en cas de signaux stationnaires. Cela améliore la performance en présence de bruit de fond stationnaire.

Le facteur de verbalisation est donné par $\lambda = 0,5(1-r_v)$ avec $r_v = (E_v - E_c)/(E_v + E_c)$, où E_v et E_c sont respectivement les énergies du vecteur de code tonal normalisé et vecteur de code innovatif normalisé. Noter que, comme la valeur de r_v est entre -1 et 1 , la valeur de λ est entre 0 et 1 . Noter que le facteur λ est associé à la grandeur de non-verbalisation avec une valeur de 0 pour les segments purement voisés et une valeur de 1 pour segments purement non voisés.

Un facteur de stabilité θ est calculé sur la base d'une mesure de distance entre les filtres LP adjacents. Ici, le facteur θ est associé à la mesure de distance des paires ISP et est limité par $0 \leq \theta \leq 1$, avec les plus grandes valeurs de θ correspondant aux signaux les plus stables.

Finalement, a facteur de lissage du gain S_m est donné par:

$$S_m = \lambda\theta \quad (64)$$

La valeur de S_m s'approche de 1 pour des signaux non voisés et stables, ce qui est le cas de signaux de bruit de fond stationnaire. Pour les signaux purement voisés ou pour les signaux instables, la valeur de S_m s'approche de 0 .

Un gain initial modifié g_0 est calculé par comparaison du gain du répertoire fixe \hat{g}_c à un seuil donné par le gain initial modifié de la sous-trame précédente, g_{-1} . Si \hat{g}_c est supérieur ou égal à g_{-1} , ensuite g_0 est calculé par decrementing \hat{g}_c par $1,5$ dB limitée par $g_0 \geq g_{-1}$. Si \hat{g}_c est plus petit que g_{-1} , alors g_0 est calculé par un accroissement de \hat{g}_c de $1,5$ dB limité par $g_0 \leq g_{-1}$.

Finalement, le gain est mis à jour avec la valeur du gain lissé comme suit:

$$\hat{g}_c = \theta g_0 + (1 + \theta)\hat{g}_c \quad (65)$$

- 7) **renforceur de délai tonal**: une procédure de renforcement du délai tonal modifie l'excitation totale $u(n)$ par filtrage de l'excitation contenue dans le répertoire fixe dans un filtre d'innovation dont la réponse en fréquence accentue les fréquences supérieures et dont les coefficients sont associés à la périodicité acheminée par le signal. Un filtre sera de la forme:

$$F_{immo}(z) = -c_{pe}z + 1 - c_{pe}z^{-1} \quad (66)$$

où $c_{pe} = 0,125(1-r_v)$, avec $r_v = (E_v - E_c)/(E_v + E_c)$ comme décrit ci-dessus. Le vecteur de code filtré par répertoire fixe est donné par:

$$c'(n) = c(n) - c_{pe}(c(n+1) + c(n-1)) \quad (67)$$

et l'excitation mise à jour est donnée par:

$$u(n) = \hat{g}_p v(n) + \hat{g}_c c'(n) \quad (68)$$

La procédure ci-dessus peut être appliquée en un seul échelon par mise à jour de l'excitation comme suit:

$$u(n) = u(n) - \hat{g}_c c_{pe} (c(n+1) + c(n-1)) \quad (69)$$

- 8) **post-traitement des éléments d'excitation (mode 6,60 kbit/s mode):** une procédure de post-traitement des éléments d'excitation est appliquée à l'excitation totale $u(n)$ par accentuation de la contribution du vecteur de répertoire adaptatif:

$$\hat{u}(n) = \begin{cases} u(n) + 0,25\beta\hat{g}_p v(n), & \hat{g}_p > 0.5 \\ u(n), & \hat{g}_p \leq 0.5 \end{cases} \quad (70)$$

la commande de gain adaptative (AGC, *adaptative gain control*) est utilisée afin de compenser la différence de gain entre la non-excitation accentuée $u(n)$ et l'excitation accentuée $\hat{u}(n)$. Le facteur de normalisation par le gain η pour l'excitation accentuée est calculé par:

$$\eta = \begin{cases} \sqrt{\frac{\sum_{n=0}^{63} u^2(n)}{\sum_{n=0}^{63} \hat{u}^2(n)}}, & \hat{g}_p > 0.5 \\ 1.0, & \hat{g}_p \leq 0.5 \end{cases} \quad (71)$$

le signal d'excitation normalisé par le gain accentué $\hat{u}'(n)$ est donné par:

$$\hat{u}'(n) = \hat{u}(n)\eta \quad (72)$$

le signal vocal reconstruit pour la sous-trame de longueur 64 est donné par

$$\hat{s}(n) = \hat{u}'(n) - \sum_{i=1}^{16} \hat{a}_i \hat{s}(n-i), \quad n = 0, \dots, 63 \quad (73)$$

où \hat{a}_i sont les coefficients interpolés du filtre LP.

Le signal vocal de synthèse $\hat{s}(n)$ est ensuite soumis à un post-traitement adaptatif qui est décrit dans le paragraphe suivant.

6.2 Filtrage passe-haut, renormalisation multiplicatrice et interpolation

Le filtre passe-haut sert de précaution contre les composants parasites à basse fréquence. Le signal est filtré dans le filtre passe-haut $H_{h1}(z)$ et dans le filtre désaccentué $H_{de_emph}(z)$.

Finalement, le signal est suréchantillonné à 16 kHz afin d'obtenir le signal de synthèse en bande inférieure $\hat{s}_{16k}(n)$. $\hat{s}_{16k}(n)$ est produit d'abord par suréchantillonnage à 5 fois du signal de synthèse en bande inférieure $\hat{s}_{12,8k}(n)$ à 12,8 kHz, ensuite par filtrage de la sortie dans $H_{decim}(z)$, et finalement par sous-échantillonnage à 4 fois du signal.

(La normalisation multiplicatrice consiste en une multiplication par 2 de la sortie du filtrage passe-haut afin de compenser la normalisation réductrice dans la phase de prétraitement.)

6.3 Bande des hautes fréquences

Dans la bande de fréquence supérieure (6,4-7,0 kHz), l'excitation est produite de façon à modéliser les fréquences les plus élevées. Le contenu en hautes fréquences est produit par remplissage de la partie supérieure du spectre avec un bruit blanc convenablement échelonné dans le domaine d'excitation, converti ensuite dans domaine vocal par mise en forme dans un filtre issu du filtre LP de synthèse utilisé pour synthétiser le signal sous-échantillonné.

6.3.1 Production de l'excitation à haute fréquence

L'excitation en bande haute est obtenue d'abord par production d'un bruit blanc $u_{HB1}(n)$. La puissance de l'excitation en bande haute est rendue égale à la puissance de l'excitation en bande inférieure $u_2(n)$, ce qui signifie que:

$$u_{HB2}(n) = u_{HB1}(n) \sqrt{\frac{\sum_{k=0}^{63} u_2^2(k)}{\sum_{k=0}^{63} u_{HB1}^2(k)}} \quad (74)$$

Finalement, l'excitation en bande haute est trouvée par:

$$u_{HB}(n) = \hat{g}_{HB} u_{HB2}(n) \quad (75)$$

où \hat{g}_{HB} est un facteur de gain.

Dans le mode 23,85 kbit/s, \hat{g}_{HB} est décodé à partir de l'indice de gain reçu.

Dans les modes 6,60, 8,85, 12,65, 14,25, 15,85, 18,25, 19,85 et 23,05 kbit/s, g_{HB} est estimé au moyen d'informations de verbalisation limitées par $[0,1, 1,0]$. L'écart de niveau à court terme de la synthèse $e_{\text{écart de niveau à court terme}}$ est d'abord calculé

$$e_{\text{tilt}} = \frac{\sum_{n=1}^{63} \hat{s}_{hp}(n) \hat{s}_{hp}(n-1)}{\sum_{n=0}^{63} \hat{s}_{hp}^2(n)} \quad (76)$$

où $\hat{s}_{hp}(n)$ est la synthèse vocale en bande inférieure filtrée en passe-haut $\hat{s}_{12.8k}(n)$ avec fréquence de coupure de 400 Hz. Le gain \hat{g}_{HB} est ensuite trouvé par

$$g_{HB} = w_{SP} g_{SP} + (1 - w_{SP}) g_{BG} \quad (77)$$

où $g_{SP} = 1 - e_{\text{écart de niveau à court terme}}$ est le gain pour le signal vocal, $g_{BG} = 1,25g_{SP}$ est le gain pour le signal de bruit de fond, et w_{SP} est une fonction de pondération mise à 1, lorsque la détection VAD est activée, et mise à 0 lorsque la détection VAD est désactivée. Le gain g_{HB} est limité entre $[0,1, 1,0]$. S'il y a des segments voisés où moins d'énergie est présente aux hautes fréquences, l'écart $e_{\text{écart de niveau à court terme}}$ s'approche de 1 ce qui se traduit par un gain inférieur g_{HB} . Cela réduit l'énergie du bruit produit en cas de segments voisés.

6.3.2 Filtre de prédiction linéaire pour la bande des hautes fréquences

6.3.2.1 Mode 6,60 kbit/s

Le filtre LP de synthèse en bande haute $A_{HB}(z)$ est trouvé par extrapolation du vecteur quantifié de fréquence ISF \mathbf{f} en vecteur de fréquence ISF \mathbf{f}_e du 2^e ordre. L'on obtient d'abord le maximum de l'autocorrélation $C_{\max}(i)$ du vecteur de différence vectorielle de fréquence ISF $f_{\Delta}(i) = f(i+1) - f(i), i = 1, \dots, 14$. Ensuite un nouveau vecteur de fréquence ISF de 16 kHz $f'_e(i)$ est calculé par:

$$f'_e(i) = \begin{cases} f(i-1), & i = 1, \dots, 15 \\ f'_e(i-1) + f'_e(i - C_{\max}(i) - 1) - f'_e(i - C_{\max}(i) - 2), & i = 16, \dots, 19 \end{cases} \quad (78)$$

Une approximation du dernier élément d'un nouveau vecteur de fréquence ISF f_{e19} est mise à jour sur la base de coefficients de fréquence inférieure. Le nouveau vecteur de différence vectorielle de fréquence ISF $f'_{e\Delta}(i)$ est:

$$f'_{e\Delta}(i) = c_{\text{scale}}(f'_e(i) - f'_e(i-1)), \quad i = 16, \dots, 19 \quad (79)$$

où c_{scale} effectue une normalisation de $f'_{e\Delta}(i)$ telle que $f_e(19)$ seront égal à f_{e19} . Afin d'assurer la stabilité, $f'_{e\Delta}(i)$ est limitée par:

$$f'_{e\Delta}(i) + f'_{e\Delta}(i-1) > 500, \quad i = 17, \dots, 19 \quad (80)$$

Finalement, le vecteur de fréquence ISF extrapolé \mathbf{f}_e est obtenu par:

$$f_e(i) = \begin{cases} f(i), & i = 1, \dots, 15 \\ f'_{e\Delta}(i) + f_e(i-1), & i = 16, \dots, 19 \\ f(16), & i = 20 \end{cases} \quad (81)$$

\mathbf{f}_e est converti dans le domaine cosinusoidal afin d'obtenir \mathbf{q}_e avec une fréquence d'échantillonnage de 16 000 Hz. Le filtre LP de synthèse en bande haute $A_{HB}(z)$ est obtenu par conversion de \mathbf{q}_e en filtre LP comme décrit au 5.2.4 avec $m = 20$.

6.3.2.2 Modes 8,85, 12,65, 14,25, 15,85, 18,25, 19,85, 23,05 ou 23,85 kbit/s

Le filtre LP de synthèse en bande haute $A_{HB}(z)$ est le filtre LP de synthèse pondéré en bande inférieure:

$$A_{HB}(z) = \hat{A}(z/0.8) \quad (82)$$

où $\hat{A}(z)$ est le filtre LP de synthèse interpolé. $\hat{A}(z)$ a été calculé par analyse du signal avec la fréquence d'échantillonnage de 12,8 kHz mais il est utilisé pour le signal de 16 kHz. Pratiquement, cela signifie que la réponse en fréquence $FR_{16}(f)$ de $A_{HB}(z)$ est obtenue par:

$$FR_{16}(f) = FR_{12,8}\left(\frac{12,8}{16}f\right) \quad (83)$$

où $FR_{12,8}(f)$ est la réponse en fréquence de $A(z)$. Cela signifie que la bande 5,1-5,6 kHz dans le domaine de 12,8 kHz seront mappés à 6,4-7,0 kHz dans le domaine de 16 kHz.

6.3.3 Synthèse de la bande des hautes fréquences

$u_{HB}(n)$ est filtré dans $A_{HB}(z)$. La sortie de cette synthèse en bande haute $s_{HB}(n)$ est filtrée dans un filtre FIR passe-bande $H_{HB}(z)$ qui possède une bande passante de 6 à 7 kHz. Finalement, s_{HB} est ajouté à au signal vocal synthétisé \hat{s}_{16k} afin de produire le signal vocal de sortie synthétisé $\hat{s}_{\text{output}}(n)$.

7 Répartition détaillée des bits du codec adaptatif à multidébit en large bande

La répartition détaillée des bits dans le codeur vocal adaptatif à multidébit en bande large est indiquée pour chaque mode dans les Tableaux 12a à 12i. Ces tableaux montrent l'ordre des bits produit par le codeur vocal. Noter que le bit de poids fort (MSB) de chaque paramètre du codec est toujours envoyé en premier.

Tableau 12a/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 477 bits/20 ms, mode 23,85 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s68	Indice de répertoire 1 pour piste 1
s69-s79	Indice de répertoire 1 pour piste 2
ss80 –s90	Indice de répertoire 1 pour piste 3
s91–s101	Indice de répertoire 1 pour piste 4
s102–s112	Indice de répertoire 2 pour piste 1
s113–s123	Indice de répertoire 2 pour piste 2
s124-s134	Indice de répertoire 2 pour piste 3
s135-s145	Indice de répertoire 2 pour piste 4
s146-s152	Gains de répertoire
s153-s156	Energie en bande haute
Sous-trame 2	
s157-s162	Indice du répertoire adaptatif (relatif)
s163-s262	Même description que s57-s156
Sous-trame 3	
s263-s371	Même description que s48-s156
Sous-trame 4	
s372-s477	Même description que s157-s262

Tableau 12b/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 461 bits/20 ms, mode 23,05 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 3 ^e sous-vecteur de paire ISP
s31-s37	Indice du 4 ^e sous-vecteur de paire ISP
s38-s42	Indice du 5 ^e sous-vecteur de paire ISP
s43-s47	Indice du 6 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s68	Indice de répertoire 1 pour piste 1
s69-s79	Indice de répertoire 1 pour piste 2
ss80 –s90	Indice de répertoire 1 pour piste 3
s91–s101	Indice de répertoire 1 pour piste 4
s102–s112	Indice de répertoire 2 pour piste 1
s113–s123	Indice de répertoire 2 pour piste 2
s124-s134	Indice de répertoire 2 pour piste 3
s135-s145	Indice de répertoire 2 pour piste 4
s146-s152	Gains de répertoire
Sous-trame 2	
s153-s158	Indice du répertoire adaptatif (relatif)
s159-s254	Même description que s57-s152
Sous-trame 3	
s255-s359	Même description que s48-s152
Sous-trame 4	
s360-s461	Même description que s153-s254

Tableau 12c/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 397 bits/20 ms, mode 19,85 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s67	Indice de répertoire 1 pour piste 1
s68-s77	Indice de répertoire 1 pour piste 2
s78-s79	Selecteur d'impulsion pour piste 3
s80-s81	Selecteur d'impulsion pour piste 4
s82-s91	Indice de répertoire 2 pour piste 1
s92-s101	Indice de répertoire 2 pour piste 2
s102-s115	Indice de répertoire pour piste 3
s116-s129	Indice de répertoire pour piste 4
s130-s136	VQ gain
Sous-trame 2	
s137-s142	Indice du répertoire adaptatif (relatif)
s143-s222	Même description que s57-s136
Sous-trame 3	
s223-s311	Même description que s48-s136
Sous-trame 4	
s312-s397	Même description que s137-s222

Tableau 12d/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 365 bits/20 ms, mode 18,25 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s59	Selecteur d'impulsion pour piste 1
s60-s61	Selecteur d'impulsion pour piste 2
s62-s63	Selecteur d'impulsion pour piste 3
s64-s65	Selecteur d'impulsion pour piste 4
s66-s79	Indice de répertoire pour piste 1
s80-s93	Indice de répertoire pour piste 2
s94-s107	Indice de répertoire pour piste 3
s108-s121	Indice de répertoire pour piste 4
s122-s128	VQ gain
Sous-trame 2	
s129-s134	Indice du répertoire adaptatif (relatif)
s135-s206	Même description que s57-s128
Sous-trame 3	
s207-s287	Même description que s48-s128
Sous-trame 4	
s288-s365	Même description que s129-s206

Tableau 12e/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 317 bits/20 ms, mode 15,85 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s70	Indice de répertoire pour piste 1
s71-s83	Indice de répertoire pour piste 2
s84-s96	Indice de répertoire pour piste 3
s97-s109	Indice de répertoire pour piste 4
s110-s116	VQ gain
Sous-trame 2	
s117-s122	Indice du répertoire adaptatif (relatif)
s123-s182	Même description que s57-s116
Sous-trame 3	
s183-s251	Même description que s48-s116
Sous-trame 4	
s252-s317	Même description que s117-s182

Tableau 12f/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 285 bits/20 ms, mode 14,25 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s70	Indice de répertoire pour piste 1
s71-s83	Indice de répertoire pour piste 2
s84-s92	Indice de répertoire pour piste 3
s93-s101	Indice de répertoire pour piste 4
s102-s108	VQ gain
Sous-trame 2	
s109-s114	Indice du répertoire adaptatif (relatif)
s115-s166	Même description que s57-s108
Sous-trame 3	
s167-s227	Même description que s48-s108
Sous-trame 4	
s228-s285	Même description que s109-s166

Tableau 12g/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 253 bits/20 ms, mode 12,65 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s56	Indice du répertoire adaptatif
s57	Fanion de filtrage LTP
s58-s66	Indice de répertoire pour piste 1
s67-s75	Indice de répertoire pour piste 2
s76-s84	Indice de répertoire pour piste 3
s85-s93	Indice de répertoire pour piste 4
s94-s100	VQ gain
Sous-trame 2	
s101-s106	Indice du répertoire adaptatif (relatif)
s107-s150	Même description que s57-s100
Sous-trame 3	
s151-s203	Même description que s48-s100
Sous-trame 4	
s204-s253	Même description que s101-s150

Tableau 12h/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 177 bits/20 ms, mode 8,85 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s23	Indice du 3 ^e sous-vecteur de paire ISP
s24-s30	Indice du 4 ^e sous-vecteur de paire ISP
s31-s37	Indice du 5 ^e sous-vecteur de paire ISP
s38-s42	Indice du 6 ^e sous-vecteur de paire ISP
s43-s47	Indice du 7 ^e sous-vecteur de paire ISP
Sous-trame 1	
s48-s55	Indice du répertoire adaptatif
s56-s60	Indice de répertoire pour piste 1
s61-s65	Indice de répertoire pour piste 2
s66-s70	Indice de répertoire pour piste 3
s71-s75	Indice de répertoire pour piste 4
s76-s81	VQ gain
Sous-trame 2	
s82-s86	Indice du répertoire adaptatif (relatif)
s87-s112	Même description que s56-s81
Sous-trame 3	
s113-s146	Même description que s48-s81
Sous-trame 4	
s147-s177	Même description que s82-s112

Tableau 12i/G.722.2 – Paramètres de sortie du codeur de source dans l'ordre d'occurrence et répartition des bits à l'intérieur de la trame vocale de 132 bits/20 ms, mode 6,60 kbit/s

Bit (MSB-LSB)	Description
s1	Fanion VAD
s2-s9	Indice du 1 ^{er} sous-vecteur de paire ISP
s10-s17	Indice du 2 ^e sous-vecteur de paire ISP
s18-s24	Indice du 3 ^e sous-vecteur de paire ISP
s25-s31	Indice du 4 ^e sous-vecteur de paire ISP
s32-s37	Indice du 5 ^e sous-vecteur de paire ISP
Sous-trame 1	
s38-s45	Indice du répertoire adaptatif
s46 –57	Indice de répertoire
s58-s63	VQ gain
Sous-trame 2	
s64-s68	Indice du répertoire adaptatif (relatif)
s69-s86	Même description que s46-s63
Sous-trame 3	
s87-s109	Même description que s64-s86
Sous-trame 4	
s110-s132	Même description que s64-s86

8 Séquences de retour à l'état d'origine

8.1 Description fonctionnelle

Le codec vocal adaptatif multidébit à large bande est décrit en arithmétique exacte au bit près afin de faciliter son homologation ainsi que les essais généraux de son bon fonctionnement.

La réponse du codec à une séquence d'entrée prédéfinie ne peut être prévue que si les variables d'état interne du codec sont dans un état prédéfini au début de l'expérience. Le codec doit donc être mis dans ce qu'on appelle un "état d'origine" avant qu'un essai au bit près puisse être effectué. L'on procède à cette fin à une réinitialisation (procédure par laquelle les variables d'état interne du codec sont mises à leurs valeurs initiales définies). Le mode du codec (codeur/décodeur de signaux vocaux) doit être réglé par des moyens externes, lors de la réinitialisation, sur le mode d'essai correspondant.

Afin de permettre une réinitialisation de codecs distants, des trames spéciales de retour à l'état d'origine ont été définies pour le codeur et pour le décodeur, ce qui permet d'effectuer le retour à l'origine d'un codec par signalisation dans la bande.

La procédure de retour du codec à son état d'origine est définie de façon que, dans les deux sens (codeur ou décodeur), les fonctions de retour soient appelées après le traitement de la trame correspondante. La sortie correspondant à la première trame de retour dépend donc du mode de codec utilisé et de l'état du codec lors de la réception de cette trame. Elle n'est donc pas habituellement connue. La réponse du codeur à toute autre trame de retour est par définition la trame de retour correspondante du décodeur pour le mode de codec choisi. La réponse du décodeur à toute autre trame de retour est par définition la trame de retour du codeur. Cette procédure permet le retour à l'état d'origine du codeur et du décodeur à partir d'une extrémité ou de l'autre, si une configuration de bouclage est implémentée, compte tenu du verrouillage de trames approprié.

8.2 Définitions

Trame de retour du codeur: la trame de retour du codeur à son état d'origine se compose de 320 échantillons identiques, d'une longueur de 13 bits chacun, dont le bit de poids faible est mis à "un" et tous les autres bits à "zéro". Lorsqu'ils sont écrits avec une justification à gauche dans des mots de 16 bits, les échantillons ont une valeur de 0008 hex. Le décodeur de signaux vocaux doit produire cette trame en réponse à la deuxième trame et à toute autre trame de retour du décodeur si au moins deux trames de retour du décodeur ont été injectées consécutivement dans le décodeur. La trame de retour du codeur est identique pour tous les modes du codec.

Trame de retour du décodeur: il existe neuf trames différentes de retour du décodeur à son état d'origine, qui correspondent aux neuf modes du codec AMR-WB. Quand on utilise l'un de ces modes de codec, la trame de retour du décodeur correspondante est la réponse naturelle du codeur vocal à la deuxième et à toute autre trame de retour du codeur si au moins deux trames de retour du codeur ont été injectées consécutivement dans le codeur. L'Annexe C indique les valeurs paramétriques pour chaque trame de retour du décodeur à son état d'origine.

8.3 Retour à l'état d'origine du codeur

Chaque fois que le codeur vocal adaptatif multidébit à large bande reçoit en entrée une trame de retour de codeur exactement alignée sur sa segmentation interne des trames vocales, les événements suivants interviennent:

étape 1

le codeur vocal effectue ses opérations normales, y compris la détection VAD et la référence SCR puis produit, conformément au mode de codec utilisé, une trame de paramètre vocal qui est en général inconnue. Mais si le codeur vocal était dans son état d'origine au début de cette trame, la trame de paramètre vocal résultante est identique à cette trame de retour du décodeur, qui correspond au mode de codec utilisé (c'est la façon dont les trames de retour du décodeur à son état d'origine ont été construites);

étape 2

après l'achèvement normal de cette opération, le codeur vocal applique les fonctions de retour à l'état d'origine de tous les sous-modules y compris la détection VAD et la référence SCR. Puis il ramène toutes les variables d'état à leur état d'origine. Dès réception de la trame d'entrée suivante, le codeur vocal reprend à partir de son état d'origine.

NOTE – L'application d'une séquence de N trames de retour de codeur provoquera au moins l'apparition de N-1 trames de retour de décodeur à la sortie du codeur vocal.

8.4 Retour à l'état d'origine du décodeur

Chaque fois que le décodeur vocal reçoit en entrée une trame de retour de décodeur qui correspond au mode de codec utilisé, les événements suivants interviennent:

étape 1

le décodeur vocal effectue ses opérations normales et produit à sa sortie une trame vocale qui est en général inconnue. Mais si le décodeur vocal était dans son état d'origine au début de cette trame, la trame vocale résultante est remplacée par la trame de retour du codeur. Cela ne sera évidemment pas le cas mais cette hypothèse est imposée par la présente définition;

étape 2

après l'achèvement normal de cette opération, le décodeur vocal applique les fonctions de retour à l'état d'origine de tous les sous-modules y compris le générateur de bruit de confort. Puis il ramène toutes les variables d'état à leur état d'origine. Dès réception de la trame d'entrée suivante, le décodeur vocal reprend à partir de son état d'origine.

NOTE 1 – L'application d'une séquence de N trames de retour de décodeur provoquera au moins l'apparition de N-1 trames de retour de codeur à la sortie du décodeur vocal.

NOTE 2 – Par définition, la première trame de chaque séquence d'essai du décodeur doit différer au moins de la trame de retour de décodeur dans une seule position binaire des paramètres pour le codage LPC et la première sous-trame. Si le décodeur se trouve dans son état d'origine, il suffit donc de vérifier ces paramètres afin de détecter une trame de retour de décodeur subséquente. Cette définition est formulée afin de prendre en charge une implémentation à optimisation du délai.

9 Détecteur d'activité vocale (VAD, *voice activity detector*)

La fonction de l'algorithme de détection VAD est d'indiquer si chaque trame de 20 ms contient des signaux qu'il convient de transmettre, par exemple des paroles, de la musique ou des tonalités d'information. La sortie de l'algorithme de détection VAD est un fanion booléen (`VAD_flag`) indiquant la présence de tels signaux. Ce fanion est utilisé dans le codeur vocal AMR-WB et décrit dans l'Annexe B.

9.1 Symboles du détecteur VAD

Dans le cadre du présent paragraphe, les symboles suivants sont applicables.

9.1.1 Variables du détecteur VAD

<code>bckr_est[n]</code>	estimation du bruit de fond dans la bande de fréquence "n"
<code>burst_count</code>	longueur des comptages d'un segment d'activité vocale, utilisé par l'addition d'un traînage de détection VAD
<code>hang_count</code>	compteur de traînages, utilisé par l'addition d'un traînage de détection VAD
<code>level[n]</code>	niveau de signal dans la bande de fréquence "n"
<code>new_speech</code>	pointeur du codeur vocal, désignant un tampon contenant les derniers échantillons reçus d'une trame vocale
<code>noise_level</code>	niveau de bruit estimé
<code>pow_sum</code>	puissance d'entrée
<code>s(i)</code>	échantillons de la trame d'entrée
<code>snr_sum</code>	mesure entre la trame d'entrée et l'estimation de bruit
<code>speech_level</code>	niveau de parole estimé
<code>stat_count</code>	compteur d'états stationnaires
<code>stat_rat</code>	mesure indiquant un état stationnaire de la trame d'entrée
<code>tone_flag</code>	fanion indiquant la présence d'une tonalité
<code>vad_thr</code>	seuil de détection VAD
<code>VAD_flag</code>	fanion booléen de détection VAD
<code>vadreg</code>	décision intermédiaire de détection VAD

9.1.2 Constantes du détecteur VAD

<code>ALPHA_UP1</code>	constante pour la mise à jour de l'estimation de bruit
<code>ALPHA_DOWN1</code>	constante pour la mise à jour de l'estimation de bruit
<code>ALPHA_UP2</code>	constante pour la mise à jour de l'estimation de bruit

ALPHA_DOWN2	constante pour la mise à jour de l'estimation de bruit
ALPHA3	constante pour la mise à jour de l'estimation de bruit
ALPHA4	constante pour la mise à jour du niveau de signal moyen
ALPHA5	constante pour la mise à jour du niveau de signal moyen
BURST_HIGH	constante pour le contrôle de l'addition d'un traînage de détection VAD
BURST_P1	constante pour le contrôle de l'addition d'un traînage de détection VAD
BURST_SLOPE	constante pour le contrôle de l'addition d'un traînage de détection VAD
COEFF3	coefficient pour la batterie de filtres
COEFF5_1	coefficient pour la batterie de filtres
COEFF5_2	coefficient pour la batterie de filtres
HANG_HIGH	constante pour le contrôle de l'addition d'un traînage de détection VAD
HANG_LOW	constante pour le contrôle de l'addition d'un traînage de détection VAD
HANG_P1	constante pour le contrôle de l'addition d'un traînage de détection VAD
HANG_SLOPE	constante pour le contrôle de l'addition d'un traînage de détection VAD
TRAME_LEN	longueur d'une trame vocale, 256 échantillons (20 ms)
MIN_SPEECH_LEVEL1	constante pour l'estimation vocale
MIN_SPEECH_LEVEL2	constante pour l'estimation vocale
MIN_SPEECH_SNR	constante pour l'adaptation du seuil de détection VAD
NO_P1	constante pour l'adaptation du seuil de détection VAD
NO_SLOPE	constante pour l'adaptation du seuil de détection VAD
BRUIT_MAX	valeur maximale pour l'estimation de bruit
BRUIT_MIN	valeur minimale pour l'estimation de bruit
POW_TONE_THR	seuil pour la détection de tonalité
SP_ACTIVITY_COUNT	constante pour l'estimation vocale
SP_ALPHA_DOWN	constante pour l'estimation vocale
SP_ALPHA_UP	constante pour l'estimation vocale
SP_CH_MAX	constante pour l'adaptation du seuil de détection VAD
SP_CH_MIN	constante pour l'adaptation du seuil de détection VAD
SP_EST_COUNT	constante pour l'estimation vocale
SP_P1	constante pour l'adaptation du seuil de détection VAD
SP_SLOPE	constante pour l'adaptation du seuil de détection VAD

STAT_COUNT	seuil pour la détection d'un état stationnaire
STAT_THR	seuil pour la détection d'un état stationnaire
STAT_THR_level	seuil pour la détection d'un état stationnaire
THR_HIGH	constante pour l'adaptation du seuil de détection VAD
TONE_THR	seuil pour la détection de tonalité
VAD_POW_LOW	constante pour le contrôle de l'addition d'un traînage de détection VAD

9.1.3 Fonctions

+ Addition

– Soustraction

* Multiplication

/ Division

| x | valeur absolue de x

AND ET booléen

OR OU booléen

$$\sum_{n=a}^b x(n) \quad x(a) + x(a + 1) + \dots + x(b - 1) + x(b)$$

$$\text{MIN}(x, y) = \begin{cases} x, x \leq y \\ y, y < x \end{cases}$$

$$\text{MAX}(x, y) = \begin{cases} x, x \geq y \\ y, y > x \end{cases}$$

9.2 Description fonctionnelle

Le schéma fonctionnel de l'algorithme de détection VAD est décrit dans la Figure 4. L'algorithme de détection VAD utilise les paramètres du codeur vocal afin de calculer le fanion booléen de détection VAD (`VAD_flag`). Cette trame d'entrée pour détection VAD est échantillonnée à la fréquence de 12,8 kHz de sorte qu'elle contient 256 échantillons. Les échantillons de la trame d'entrée (`s(i)`) sont divisés en sous-bandes et le niveau du signal (`level[n]`) dans chaque bande est calculé. Les gains tonaux normalisés en boucle ouverte sont l'entrée pour la fonction de détection de tonalité, gains qui sont calculés par analyse du délai tonal en boucle ouverte du codeur vocal. La fonction de détection de tonalité calcule un fanion (`tone_flag`) qui indique la présence d'une tonalité de signalisation, d'un signal vocal voisé, ou un autre signal à périodicité élevée. Le niveau de bruit de fond (`bckr_est[n]`) est estimé dans chaque bande sur la base de la décision de détection VAD, de la stationnarité du signal et du fanion de tonalité. La décision intermédiaire de détection VAD est calculée par comparaison du rapport SNR d'entrée SNR (`level[n]/bckr_est[n]`) à un seuil adaptatif. Le seuil est adapté sur la base de bruit et estimations vocales à long terme. Finalement, le fanion VAD est calculé par addition du traînage à la décision intermédiaire de détection VAD.

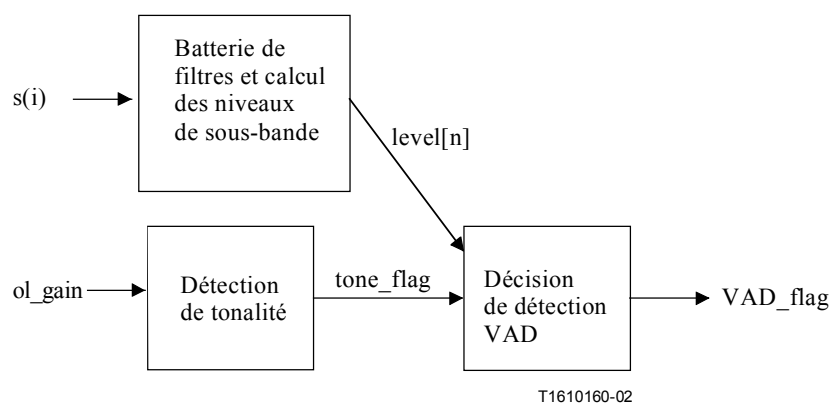


Figure 4/G.722.2 – Schéma fonctionnel simplifié de l'algorithme de détection VAD

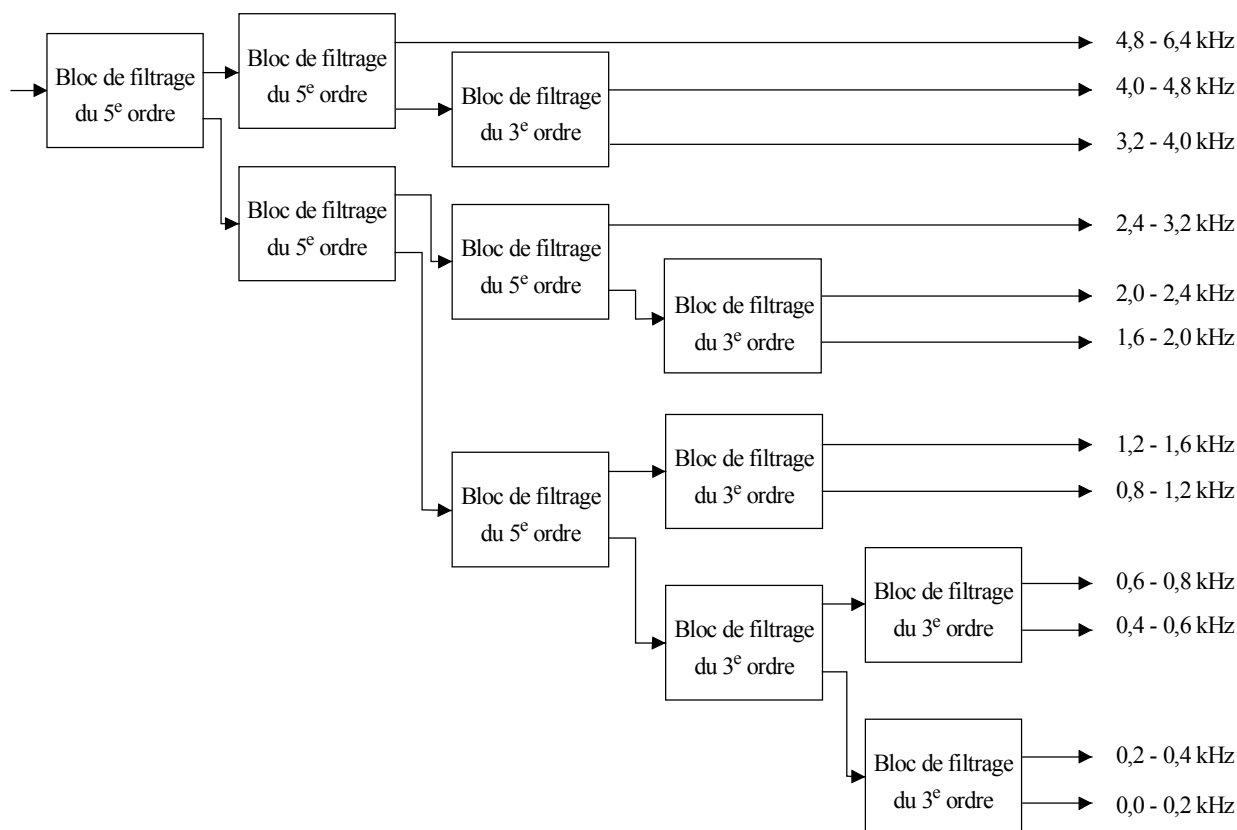
9.2.1 Batterie de filtres et calcul des niveaux de sous-bande

Le signal d'entrée est divisé en bandes de fréquence au moyen d'une batterie de 12 filtres de bande (Figure 5). Les fréquences de coupure de cette batterie de filtres sont indiquées dans le Tableau 13.

Tableau 13/G.722.2 – Fréquences de coupure pour la batterie de filtres

Numéro de bande	Fréquences
1	0-200 Hz
2	200-400 Hz
3	400-600 Hz
4	600-800 Hz
5	800-1200 Hz
6	1200-1600 Hz
7	1600-2000 Hz
8	2000-2400 Hz
9	2400-3200 Hz
10	3200-4000 Hz
11	4000-4800 Hz
12	4800-6400 Hz

L'entrée dans la batterie de filtres est une trame vocale indiquée par le pointeur de nouveau signal vocal du codeur vocal conforme à la G.722.2. Les valeurs d'entrée dans la batterie de filtres sont réduites d'un bit, ce qui garantit une normalisation sûre, c'est-à-dire qu'une saturation ne peut pas se produire pendant le calcul de la batterie de filtres.



T1610170-02

Figure 5/G.722.2 – Batterie de filtres

La batterie de filtres se compose de blocs de filtrage du 5^e et du 3^e ordre. Chaque bloc de filtrage subdivise l'entrée en passe-haut et passe-bas parties et décime la fréquence d'échantillonnage par 2. Le bloc de filtrage du 5^e ordre est calculé comme suit:

$$x_{lp}(i) = 0.5 * (A_1(x(2 * i)) + A_2(x(2 * i + 1))) \quad (84a)$$

$$x_{hp}(i) = 0.5 * (A_1(x(2 * i)) - A_2(x(2 * i + 1))) \quad (84b)$$

où

$x(i)$ signal d'entrée pour un bloc de filtrage

$x_{lp}(i)$ composante passe-bas

$x_{hp}(i)$ composante passe-haut

Le bloc de filtrage du 3^e ordre est calculé comme suit:

$$x_{lp}(i) = 0.5 * (x(2 * i + 1) + A_3(x(2 * i))) \quad (85a)$$

$$x_{hp}(i) = 0.5 * (x(2 * i + 1) - A_3(x(2 * i))) \quad (85b)$$

Les filtres $A_1()$, $A_2()$, et $A_3()$ sont des filtres passe-tout de forme directe du premier ordre filtres, dont la fonction de transfert est donnée par:

$$A(z) = \frac{C + z^{-1}}{1 + C * z^{-1}} \quad (86)$$

où C est le coefficient de filtrage.

Les coefficients pour les filtres passe-tout $A_1()$, $A_2()$, et $A_3()$ sont respectivement $COEFF5_1$, $COEFF5_2$, et $COEFF3$.

Le niveau de signal est calculé à la sortie de la batterie de filtres dans chaque la bande de fréquence comme suit:

$$level(n) = \sum_{i=START_n}^{END_n} |x_n(i)| \quad (87)$$

où:

n indice pour la bande de fréquence

$x_n(i)$ échantillon i à la sortie de la batterie de filtres dans la bande de fréquence n

$$START_n = \begin{cases} -6, & 1 \leq n \leq 4 \\ -12, & 5 \leq n \leq 8 \\ -24, & 9 \leq n \leq 11 \\ -48, & n = 12 \end{cases}$$

$$END_n = \begin{cases} 7, & 1 \leq n \leq 4 \\ 15, & 5 \leq n \leq 8 \\ 31, & 9 \leq n \leq 11 \\ 63, & n = 12 \end{cases}$$

Les indices négatifs de $x_n(i)$ se rapportent à la trame antérieure.

9.2.2 Détection de fréquence

Le but de la fonction de détection de tonalité est de détecter les tonalités d'information, les voyelles vocalisées et les autres signaux périodiques. La détection de tonalité utilise des gains tonaux normalisés en boucle ouverte (ol_gain), que sont reçus du codeur vocal. Si le gain tonal est supérieur à la constante $TONE_THR$, la tonalité est détectée et le fanion de tonalité est activé:

```
if (ol_gain > TONE_THR)
    tone_flag = 1
```

Par la recherche de délai tonal en boucle ouverte, le fanion de tonalité est calculé deux fois dans chaque trame, sauf pour le mode 6,60 kbit/s, où il est calculé seulement une fois.

9.2.3 Décision du détecteur VAD

Le schéma fonctionnel de l'algorithme de décision de détection VAD est indiqué dans la Figure 6.

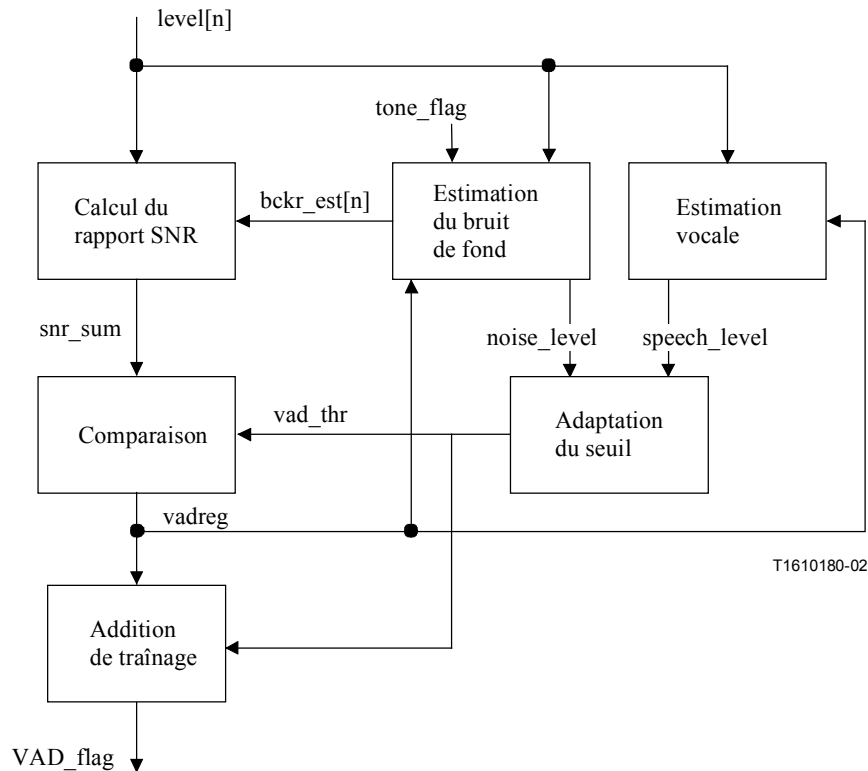


Figure 6/G.722.2 – Schéma fonctionnel simplifié de l'algorithme de décision VAD

La puissance de la trame d'entrée est calculée comme suit:

$$frame_pow = \sum_{i=0}^{FRAME_LEN} s(i) * s(i) \quad (88)$$

où les échantillons $s(i)$ de la trame d'entrée sont indiqués par le pointeur de nouveau signal `new_speech` vocal du codeur vocal. La variable `pow_sum` est la somme des puissances des trames actuelle et antérieure. Si la variable `pow_sum` est inférieure à la constante `POW_TONE_THR`, le fanion de tonalité est mis à zéro.

La différence entre les niveaux de signal de la trame d'entrée et l'estimation du bruit de fond sont calculées comme suit:

$$snr_sum = \sum_{n=1}^{12} MAX(1.0, \frac{level[n]}{bckr_est[n]})^2 \quad (89)$$

où:

`level[n]` niveau de signal dans la bande n

`bckr_est[n]` niveau d'estimation du bruit de fond dans la bande n

La décision de détection VAD est effectuée par comparaison de la variable `snr_sum` à un seuil. Le seuil (`vad_thr`) est adapté afin d'obtenir la sensibilité recherchée selon les niveaux estimés de parole et de bruit de fond.

Le niveau de bruit de fond moyen est calculé par addition des estimations de bruit dans chaque bande sauf la plus petite:

$$noise_level = \sum_{n=2}^{12} bckr_est[n] \quad (90)$$

Si le rapport SNR est inférieur au seuil (MIN_SPEECH_SNR), le niveau de parole est augmenté comme suit:

```
Si (speech_level/noise_level < MIN_SPEECH_SNR)
    Speech_level = MIN_SPEECH_SNR * noise_level
```

La valeur logarithmique pour l'estimation de bruit est calculée comme suit:

$$ilog2_noise_level = \log_2(noise_level) \quad (91)$$

Avant le calcul de la valeur logarithmique de l'estimation vocale, le niveau MIN_SPEECH_SNR*noise_level est soustrait du niveau de parole afin de corriger sa valeur dans les situations de faible SNR.

$$ilog2_speech_level = \log_2(speech_level - MIN_SPEECH_SNR * noise_level) \quad (92)$$

Le seuil pour la décision de détection VAD est calculé comme suit:

$$\begin{aligned} Vad_thr = & NO_SLOPE * (ilog2_noise_level - NO_P1) + THR_HIGH + \\ & MIN(SP_CH_MAX, MAX(SP_CH_MIN, SP_CH_MIN \\ & + SP_SLOPE * (ilog2_speech_level - SP_P1))), \end{aligned} \quad (93)$$

où NO_SLOPE, SP_SLOPE, NO_P1, SP_P1, THR_HIGH, SP_CH_MAX et SP_CH_MIN sont des constantes.

La variable vadreg indique une décision intermédiaire de détection VAD et est calculée comme suit:

```
si (snr_sum > vad_thr)
    vadreg = 1
sinon
    vadreg = 0
```

9.2.3.1 Addition d'un traînage

Avant que le fanion VAD final soit donné, un traînage est ajouté. L'addition d'un traînage aide à détecter les terminaisons à faible puissance de segment d'activité vocale, qui sont subjectivement importantes mais difficiles à détecter.

Le fanion VAD est mis à "1" si moins de hang_len trames avec décision "0" ont passé depuis la détection de burst_len décisions "1" consécutives. Les variables hang_len et burst_len sont calculées au moyen de vad_thr comme suit:

$$hang_len = MAX(HANG_LOW, (HANG_SLOPE * (vad_thr - HANG_P1) + HANG_HIGH)) \quad (94)$$

$$burst_len = BURST_SLOPE * (vad_thr - BURST_P1) + BURST_HIGH \quad (95)$$

La puissance de la trame d'entrée est comparée à un seuil (VAD_POW_LOW). Si la puissance est inférieure, le fanion VAD est mis à "0" et aucun traînage n'est ajouté. Le fanion vad_flag est calculé comme suit:

```
Vad_flag = 0;
si (pow_sum < VAD_POW_LOW)
    burst_count = 0
    hang_count = 0
sinon
    si (vadreg = 1)
        burst_count = burst_count + 1
        si (burst_count >= burst_len)
```

```

    hang_count = hang_len
    VAD_flag = 1
sinon
    burst_count = 0
    si (hang_count > 0)
        hang_count = hang_count - 1
        VAD_flag = 1

```

9.2.3.2 Estimation du bruit de fond

L'estimation du bruit de fond ($bckr_est[n]$) est mise à jour au moyen des niveaux d'amplitude de la précédente trame. La mise à jour est donc retardée d'une trame afin d'éviter le départ non détecté de segments d'activité vocale et une dégradation de l'estimation de bruit. La vitesse de mise à jour pour la trame actuelle est sélectionnée au moyen des décisions intermédiaires de détection VAD ($vadreg$) et du compteur de stationnarité ($stat_count$) comme suit:

```

si (vadreg for the last 4 frames has been zero)
    alpha_up = ALPHA_UP1
    alpha_down = ALPHA_DOWN1
sinon si (stat_count = 0)
    alpha_up = ALPHA_UP2
    alpha_down = ALPHA_DOWN2
sinon
    alpha_up = 0
    alpha_down = ALPHA3

```

La variable $stat_count$ indique un état stationnaire et son but est expliqué plus loin dans le présent paragraphe. Les variables $alpha_up$ et $alpha_down$ définissent respectivement la vitesse de mise à jour vers le haut et vers le bas. La vitesse de mise à jour dans chaque bande "n" est sélectionnée comme suit:

```

si ( $bckr\_est_m[n] < level_{m-1}[n]$ )
    alpha[n] = alpha_up
sinon
    alpha[n] = alpha_down

```

Finalement, l'estimation de bruit est mise à jour comme suit:

$$bckr_est_{m+1}[n] = (1.0 - alpha[n]) * bckr_est_m[n] + alpha[n] * level_{m-1}[n], \quad (96)$$

où:

- n est l'indice de la bande de fréquence
- m est l'indice de la trame

Le niveau de l'estimation du bruit de fond ($bckr_est[n]$) est limité par les constantes $NOISE_MIN$ et $NOISE_MAX$.

Si le niveau de bruit de fond augmente soudainement, *vadreg* sera mis à "1" et le bruit de fond n'est pas mis à jour normalement vers le haut. Afin de résoudre cette situation, la mise à jour de l'estimation du bruit de fond est activée si la décision intermédiaire de détection VAD (*vadreg*) est "1" pendant assez longtemps et si le spectre est un état stationnaire. Un état stationnaire (*stat_rat*) est estimé au moyen de l'équation:

$$stat_rat = \sum_{n=1}^{12} \frac{MAX(STAT_THR_LEVEL, MAX(ave_level_m[n], level_m[n]))}{MAX(STAT_THR_LEVEL, MIN(ave_level_m[n], level_m[n]))} \quad (97)$$

où:

STAT_THR_LEVEL: est une constante

n: indice de la bande de fréquence

m: indice de la trame

ave_level: niveau moyen du signal d'entrée

Si l'estimation d'état stationnaire (*stat_rat*) est supérieure à un seuil, le compteur de signaux stationnaires (*stat_count*) est mis à la valeur initiale définie par la constante *STAT_COUNT*. Si le signal n'est pas stationnaire mais qu'un signal vocal ait été détecté (la décision de détection VAD est "1"), la variable *stat_count* est diminuée d'une unité dans chaque trame jusqu'à ce qu'elle soit égale à zéro.

si (les 5 derniers fanions de tonalité ont eu la valeur 1)

stat_count = STAT_COUNT

sinon

if (les 8 dernières décisions internes de détection VAD ont eu la valeur zéro) OU (*stat_rat* > *STAT_THR*)

stat_count = STAT_COUNT

sinon

si (*vadreg*) AND (*stat_count* ≠ 0)

stat_count = stat_count - 1

Les niveaux de signal moyens (*ave_level[n]*) sont calculés comme suit:

$$ave_level_{m+1}[n] = (1.0 - \alpha) * ave_level_m[n] + \alpha * level_m[n] \quad (98)$$

La vitesse de mise à jour (*alpha*) est sélectionnée comme suit pour l'équation précédente:

si (*stat_count* = *STAT_COUNT*)

alpha = 1.0

sinon si (*vadreg* = 1)

alpha = ALPHA5

sinon

alpha = ALPHA4

9.2.3.3 Estimation du niveau de parole

Le niveau d'entrée en bande large est calculé par sommation des niveaux d'entrée dans chaque bande, sauf la plus petite, comme suit:

$$in_level = \sum_{n=2}^{12} level[n] \quad (99)$$

Une trame est censée contenir de la parole si son niveau est assez élevé (*MIN_SPEECH_level1*) et si le fanion de détection VAD intermédiaire (*vadreg*) est activé ou si le niveau d'entrée est

supérieur au niveau actuel de l'estimation de parole. Le niveau maximal (sp_max) de SP_EST_COUNT trames est recherché. Si le nombre $SP_ACTIVITY_COUNT$ de trames vocales est inférieur au nombre SP_EST_COUNT de trames, l'estimation du niveau de parole est mise à jour par le niveau maximal de signal (sp_max). Le pseudo-code pour l'estimation du niveau de parole est le suivant:

```

si (SP_ACTIVITY_COUNT > SP_EST_COUNT - sp_est_cnt + sp_max_cnt)
    sp_est_cnt = 0
    sp_max_cnt = 0
    sp_max = 0
sp_est_cnt = sp_est_cnt + 1
si (in_level > MIN_SPEECH_LEVEL1) AND ((vadreg = 1) OR (in_level >
speech_level))
    sp_max_cnt = sp_max_cnt + 1
    sp_max = MAX(sp_max, in_level)
    si (sp_max_cnt > SP_ACTIVITY_COUNT)
        si (sp_max > MIN_SPEECH_LEVEL2)
            if (sp_max > speech_level)
                speech_level = speech_level + SP_ALPHA_UP * (sp_max -
speech_level)
            sinon
                speech_level = speech_level + SP_ALPHA_DOWN * (sp_max -
speech_level)
        sp_max_cnt = 0
        sp_max = 0
        sp_est_cnt = 0

```

10 Bibliographie (informative)

- [2] SCHROEDER M.R. et ATAL B.S.: Code-Excited Linear Prediction (CELP): High quality speech at very low bit rates, in Proc. *ICASSP'85*, pp. 937-940, 1985.
- [3] RABINER L.R. et SCHAEFER R.W.: *Digital processing of speech signals*, Prentice-Hall Int., 1978.
- [4] ITAKURA F.: Line spectral representation of linear predictive coefficients of speech signals, *J. Acoust. Soc. Amer.*, Vol. 57, Supplement No. 1, S35, 1975.
- [5] BISTRITZ Y. et PELLERM S.: Immittance Spectral Pairs (ISP) for speech encoding, in Proc. *ICASSP'93*, pp. II-9 – II-12.
- [6] PALIWAL K.K et ATAL B.S.: Efficient vector quantization of LPC parameters at 24 bits/frame, *IEEE Trans. Speech and Audio Processing*, Vol. 1, no 1, pp. 3-14, 1993.
- [7] KABAL P. et RAMACHANDRAN R.P.: The computation of line spectral frequencies using Chebyshev polynomials, *IEEE Trans. on ASSP*, Vol. 34, no. 6, pp. 1419-1426, 1986.
- [8] JÄRVINEN K., VAINIO J., KAPANEN P., HONKANEN T., HAAVISTO P., SALAMI R., LAFLAMME C., et ADOUL J.-P.: GSM enhanced full rate speech codec, in *Proc. ICASSP'97*, pp. 771-774.
- [9] HONKANEN T., VAINIO J., JÄRVINEN K., HAAVISTO P., SALAMI R., LAFLAMME C., et ADOUL J.-P.: Enhanced full rate speech codec for IS-136 digital cellular system, in *Proc. ICASSP'97*, pp. 731-734.

- [10] HAGEN R., EKUDDEN E., JOHANSSON B., et KLEIJN W.B.: Removal of sparse-excitation artifacts in CELP, in *Proc. ICASSP'98*, pp. I-145-I-148.
- [11] 3GPP TS 26.171: AMR Wideband Speech Codec; General description.
- [12] 3GPP TS 26.190: AMR Wideband Speech Codec; Transcoding functions.
- [13] 3GPP TS 26.173: AMR Wideband Speech Codec; ANSI-C code.
- [14] 3GPP TS 26.174: AMR Wideband Speech Codec; Test sequences.
- [15] 3GPP TS 26.193: AMR Wideband Speech Codec; Source Controlled Rate operation.
- [16] 3GPP TS 26.194: AMR Wideband Speech Codec; Voice Activity Detector (VAD).
- [17] 3GPP TS 26.192: AMR Wideband Speech Codec; Comfort Noise Aspects.
- [18] 3GPP TS 26.191: AMR Wideband Speech Codec; Error Concealment of Lost Frames.
- [19] 3GPP TS 26.201: AMR Wideband Speech Codec; Frame Structure.
- [20] 3GPP TR 26.901: AMR Wideband Speech Codec; Feasibility study report.

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	RGT et maintenance des réseaux: systèmes de transmission, circuits téléphoniques, télégraphie, télécopie et circuits loués internationaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux de données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information et protocole Internet
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication