



(11) **EP 1 836 699 B1**

(12) **FASCICULE DE BREVET EUROPEEN**

(45) Date de publication et mention de la délivrance du brevet:  
**29.06.2011 Bulletin 2011/26**

(21) Numéro de dépôt: **06709052.2**

(22) Date de dépôt: **09.01.2006**

(51) Int Cl.:  
**G10L 19/08 (2006.01)**

(86) Numéro de dépôt international:  
**PCT/FR2006/000038**

(87) Numéro de publication internationale:  
**WO 2006/075078 (20.07.2006 Gazette 2006/29)**

(54) **Procédé et dispositif de codage audio optimisé entre deux modèles de prediction à long terme**

Verfahren und Vorrichtung zur Ausführung einer optimalisierten Audiokodierung zwischen zwei Langzeitvorhersagemodellen

Method and device for carrying out optimized audio coding between two long-term prediction models

(84) Etats contractants désignés:  
**AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IS IT LI LT LU LV MC NL PL PT RO SE SI SK TR**

(30) Priorité: **11.01.2005 FR 0500272**

(43) Date de publication de la demande:  
**26.09.2007 Bulletin 2007/39**

(73) Titulaire: **FRANCE TELECOM**  
**75015 Paris (FR)**

(72) Inventeurs:  
• **GHENANIA, Mohamed**  
**72100 Le Mans (FR)**  
• **LAMBLIN, Claude**  
**22700 Perros Guirec (FR)**

(74) Mandataire: **Cabinet Plasseraud**  
**52, rue de la Victoire**  
**75440 Paris Cedex 09 (FR)**

(56) Documents cités:  
**WO-A-03/058407 WO-A-2005/066936**

- **GHENANIA M ET AL: "TRANSCODAGE INTELLIGENT A FAIBLE COMPLEXITE EXTRE LES CODEURS UIT-T G.729 ET 3GPP NB-AMR (12.2 KBIT/S)" CORESA. COMPRESSION ET REPRESENTATION DES SIGNAUX AUDIOVISUELS, 25 mai 2004 (2004-05-25), pages 85-88, XP001199662**
- **JIN-KYU CHOI ET AL: "Improvement issues on transcoding algorithms: for the flexible usage to the various pairs of speech codec" 2004 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, vol. 1, 17 mai 2004 (2004-05-17), - 21 mai 2004 (2004-05-21) pages 269-272, XP002345813 MONTREAL, QUE., CANADA ISBN: 0-7803-8484-9**

**EP 1 836 699 B1**

Il est rappelé que: Dans un délai de neuf mois à compter de la publication de la mention de la délivrance du brevet européen au Bulletin européen des brevets, toute personne peut faire opposition à ce brevet auprès de l'Office européen des brevets, conformément au règlement d'exécution. L'opposition n'est réputée formée qu'après le paiement de la taxe d'opposition. (Art. 99(1) Convention sur le brevet européen).

## Description

**[0001]** La présente invention concerne le codage/décodage en compression de signaux audio numériques, notamment de signaux de parole et/ou de signaux multimédias, notamment pour des applications de transmission ou de stockage. Elle vise plus particulièrement une détermination efficace des paramètres d'un second modèle de prédiction à long terme (ou "LTP" pour "Long Term Prediction"), à partir des paramètres d'au moins un premier modèle de prédiction LTP.

**[0002]** Les codeurs en compression utilisent des propriétés du signal audio numérique telles que sa stationnarité locale, exploitée par des filtres de prédiction à court terme, ainsi que sa structure harmonique, exploitée par des filtres de prédiction à long terme LTP. Typiquement, les sons voisés d'un signal de parole (tels que les voyelles) présentent une corrélation à long terme due à la vibration des cordes vocales. La corrélation à long terme est modélisée par un filtre LTP noté  $P(z)$  qui permet de restituer la structure harmonique en utilisant un filtre de synthèse du type :

$$H_{LT}(z) = \frac{1}{1 - P(z)}$$

**[0003]** La forme la plus simple du filtre de prédiction à long terme est le filtre  $P(z)$  à un seul coefficient  $\beta$  (appelé aussi gain) et à retard entier  $T$  tel que  $P(z) = \beta z^{-T}$ . Le retard  $T$  est aussi appelé période de "pitch", ou plus simplement "pitch".

**[0004]** Actuellement, des modélisations plus élaborées visent :

- une modélisation à plusieurs coefficients (dite "multitap") :  $P(z) = \sum_{i=-k}^k \beta_i z^{-T-i}$ ,

- ou encore une modélisation à retards multiples :  $P(z) = \sum_{i=1}^k \beta_i z^{-iT}$ ,

- ou encore une modélisation à retard fractionnaire qui utilise des sur et sous échantillonnages avec des filtres

d'interpolation :  $P(z) = \beta \sum_{i=0}^{2I-1} p_i(i) z^{-(T-I+i)}$ , où pour un retard  $(T+I/D)$ , de résolution  $1/D$ , les coefficients  $p_i$

( $i$ ) sont donnés par  $p_i(i) = h_{inter}(iD-I)$ ,  $0 \leq i \leq D-1$ ,  $h_{inter}$  étant un filtre d'interpolation de longueur  $2ID+1$ .

**[0005]** Les paramètres du filtre (retard et gain(s)) varient selon les signaux à coder et pour un même signal au cours du temps. Par exemple, en codage de parole, la plage des périodes de pitch cherche à couvrir la gamme des fréquences fondamentales de la voix humaine (des voix graves aux voix aiguës). Pour un même locuteur, cette fréquence varie aussi temporellement. De même, le(s) coefficient(s) du filtre évolue(nt) aussi dans le temps.

**[0006]** Au codage, les paramètres de  $P(z)$  sont déterminés soit par une analyse en boucle ouverte soit par une analyse en boucle fermée ou le plus souvent par une combinaison des deux analyses. L'analyse en boucle ouverte est effectuée en minimisant l'erreur de prédiction sur le signal à modéliser. L'analyse en boucle fermée (dite "analyse par synthèse") minimise l'erreur quadratique, habituellement pondérée, entre le signal vocal à modéliser et le signal de synthèse. Habituellement, une recherche en boucle ouverte est d'abord prévue afin de déterminer une première estimation du pitch appelée "pitch en boucle ouverte". Puis, une recherche à analyse par synthèse sur un voisinage restreint autour de cette valeur d'ancrage permet d'obtenir une valeur plus précise du pitch. Ces analyses sont effectuées sur des blocs d'échantillons. Les longueurs des blocs d'analyse en boucle ouverte et fermée ne sont pas nécessairement égales. Souvent, une seule analyse en boucle ouverte est effectuée pour plusieurs analyses en boucle fermée.

**[0007]** Pour tout modèle LTP (monotap ou multitap), la détermination des paramètres LTP est très coûteuse en complexité de calcul. Elle est généralement constituée d'une boucle ouverte sur un grand bloc d'échantillons suivie de boucles fermées sur plusieurs sous-blocs d'échantillons (appelés aussi sous-trames). En particulier, la recherche en boucle ouverte du délai harmonique est une opération très coûteuse, au codage. Habituellement, elle nécessite le calcul d'une fonction d'auto-corrélation du signal pour de nombreuses valeurs (en fait sur plage de variation des retards). Dans le codeur selon la norme UIT-T G.723.1, cette plage de retards comporte 125 retards entiers (de 18 à 142) et le retard en boucle ouverte est estimé toutes les 15 ms (soit donc pour des blocs de 120 échantillons). Dans le codeur selon la norme UIT-T G.729 à 8 kbit/s, l'analyse en boucle ouverte est effectuée toutes les 10 ms (à chaque bloc de 80 échantillons) et explore une plage de 124 retards entiers (de 20 à 143). Cette opération constitue près de 70% de la complexité de

l'analyse LTP pour ce type de codage.

**[0008]** Même si elle est focalisée autour du retard obtenu en boucle ouverte, la boucle fermée est aussi extrêmement coûteuse en calculs et, par conséquent, en ressources. Elle nécessite la génération d'excitations adaptatives et leur filtrage. Par exemple, dans le codage G.723.1 qui utilise un modèle LTP multitap, l'analyse en boucle fermée détermine conjointement le vecteur de gains ( $\beta_j$ ) et un délai  $\lambda$  (en tant que pitch candidat) de chaque sous-trame en explorant un dictionnaire de vecteurs de gains pour plusieurs valeurs de pitch candidates. Cette analyse constitue près de la moitié de la complexité totale du codeur G.723.1 à 5,3 kbit/s.

**[0009]** La complexité de l'analyse LTP est particulièrement critique lorsque plusieurs codages doivent être effectués par une même unité de traitement telle qu'une passerelle chargée de gérer de nombreuses communications en parallèle ou un serveur distribuant de nombreux contenus multimédias. Le problème de la complexité est encore accru par la multiplicité des formats de compression qui circulent sur les réseaux. Il est alors prévu plusieurs codages, soit en cascade (ou "*transcodage*"), soit en parallèle (codage multi-format ou codage multi-mode). Le transcodage est typiquement utilisé lorsque, dans une chaîne de transmission, une trame de signal compressée émise par un codeur ne peut plus poursuivre son chemin, sous ce format. Le transcodage permet de convertir cette trame sous un autre format compatible avec la suite de la chaîne de transmission. La solution la plus élémentaire (et la plus courante à l'heure actuelle) est la mise bout à bout d'un décodeur et d'un codeur. La trame compressée, arrivant sous un premier format, est décompressée. Ce signal décompressé est alors re-compressé sous un second format accepté par la suite de la chaîne de communication. Cette mise en cascade d'un décodeur et d'un codeur est appelée "*tandem*". Néanmoins, cette solution est très coûteuse en complexité (essentiellement à cause du recodage) et dégrade la qualité, le second codage se faisant en effet sur un signal décodé qui est une version dégradée du signal original. Par ailleurs, une trame peut rencontrer plusieurs tandems avant d'arriver à destination, ce qui accroît encore le coût en calcul et la perte de qualité. En outre, les retards liés à chaque opération de tandem s'accumulent et peuvent nuire à l'interactivité des communications.

**[0010]** Quant aux systèmes de compression multi-format où un même contenu est compressé sous plusieurs formats (typiquement dans le cas de serveurs de contenus qui diffusent un même contenu sous plusieurs formats adaptés aux conditions d'accès, de réseaux et terminaux des différents utilisateurs finaux), l'opération de multi-codage devient extrêmement complexe à mesure qu'augmente le nombre de formats désirés, ce qui peut rapidement saturer les ressources des systèmes. Un autre cas de codage multiple en parallèle est la compression multi-mode à décision a posteriori selon laquelle, à chaque segment de signal à coder, plusieurs modes de compression sont exécutés et le mode qui optimise un critère donné ou obtient le meilleur compromis débit/distorsion est sélectionné. Là encore, la complexité de chacun des modes de compression en limite le nombre et/ou conduit à élaborer une sélection a priori d'un nombre très restreint de modes.

**[0011]** Actuellement, la plupart des opérations de codage multiple ne prennent pas encore pleinement en compte les similitudes entre formats de codages, ce qui pourrait pourtant réduire la complexité et le retard algorithmique tout en limitant la dégradation apportée. Pour un même paramètre de format de codage, les différences entre codeurs résident dans la modélisation, la méthode et/ou la fréquence de calcul, ou encore la quantification.

**[0012]** De manière générale, les solutions proposées aujourd'hui s'attachent à limiter le nombre de valeurs explorées pour les paramètres d'un second modèle LTP en utilisant les paramètres choisis par le premier format, pour réduire la complexité de la recherche LTP du deuxième format.

**[0013]** Le transcodage entre deux modèles LTP monotap est le cas le plus simple. La plupart des méthodes actuellement proposées concernent le transcodage entre retards, le transcodage du gain LTP étant effectué le plus souvent au niveau du signal-même (on parle de tandem "*partiel*"). Lorsque les deux modèles sont identiques (même dictionnaire de retards et même longueur de sous-trame), une simple recopie des champs binaires des retards d'un flux de bits vers l'autre suffit. Lorsque les dictionnaires diffèrent par leur résolution (entière ou fractionnaire 1/3, 1/6, etc) et/ou par leurs plages de valeurs, un transcodage dans le domaine binaire ou des paramètres, avec une transformation éventuelle, est utilisé. La transformation peut être une quantification, une troncature, un doublement ou un dédoublement. Lorsque les longueurs de sous-frames des deux formats sont différentes, on peut prévoir une interpolation des retards. Par exemple, les retards d'un premier format recouvrant une sous-trame de sortie sont interpolés. On peut alors n'utiliser ce retard interpolé que lorsque celui-ci est proche du retard obtenu à la sous-trame précédente, sinon une recherche classique est menée. Une autre méthode plus directe, sans interpolation, consiste à sélectionner un retard parmi ces retards du premier format. Cette sélection peut être effectuée selon plusieurs critères : dernière sous-trame, sous-trame ayant le plus d'échantillons en commun avec la sous-trame du second format ou encore celle qui maximise un critère dépendant du gain LTP. Le retard déterminé est une valeur d'ancrage pour la recherche du retard du deuxième format. Il peut être utilisé comme retard en boucle ouverte du second format autour duquel est effectué une recherche en boucle fermée classique ou restreinte, ou comme une première estimation de celui-ci, ou comme ancrage d'une trajectoire de retard.

**[0014]** Dans le cas d'un transcodage entre une modélisation LTP monotap et une modélisation LTP multitap, on ne prévoit aujourd'hui qu'une mise en oeuvre simplement dans le domaine signal, du fait de la dissemblance des modélisations. La plupart des techniques de transcodage existantes se limitent à réduire la complexité de la boucle ouverte du second format en sélectionnant comme retard en boucle ouverte l'un des retards du premier format ou une interpolation

de ces retards. Toutefois, quelques techniques ont été proposées pour réduire aussi la complexité de la boucle fermée.

**[0015]** Dans le document WO-03058407, le retard fractionnaire  $\lambda'$  d'un modèle monotap est déterminé à partir du vecteur de coefficients ( $\beta_j$ ) d'un modèle multitap en calculant l'expression :

5

$$\lambda' = \lambda - \frac{\sum_{j=-2}^2 j\beta_j^2}{\sum_{j=-2}^2 \beta_j^2}$$

10

Dans le document référencé [1] :

15 "An Efficient Transcoding Algorithm For G. 723.1 And G. 729A Speech Coders", Sung-Wan Yoon, Sung-Kyo Jung, Young-Cheol Park, and Dae-Hee Youn, Proc. Eurospeech 2001, pp.2499-2502,

la recherche en boucle fermée du vecteur de gains d'un modèle multitap est restreinte à un sous-ensemble du dictionnaire des gains multitap, déterminé par le gain du modèle monotap du premier format. Cette détermination, ainsi que la composition des sous-ensembles s'effectuent comme suit : le gain global de chaque vecteur du dictionnaire de gains est calculé; puis, à partir de 170 gains globaux correspondant aux 170 vecteurs du dictionnaire, 8 sous-ensembles sont constitués et un seul de ces sous-ensembles est sélectionné en fonction du gain LTP du premier modèle monotap.

20

25 Dans une variante selon le document référencé [2] :

"Transcoding algorithm for G723.1 and AMR speech Coders: for interoperability between VoIP and Mobile Networks", Sung-Wan Yoon and al., Proc. Eurospeech 2003, pp. 1101-1104,

les sous-ensembles sont composés par apprentissage comme suit : la plage de variation du gain monotap d'un codeur NB-AMR est divisée en 8 sous-sections, puis, pour chaque sous-section, une étude statistique sur un tandem NB-AMR permet de déterminer M vecteurs de gains des dictionnaires d'un codeur selon la norme G.723.1. Ces vecteurs de gain sont statistiquement les plus probables. Le nombre M est pris égal à 40 pour le dictionnaire comportant 85 vecteurs et à 85 pour le dictionnaire comportant 170 vecteurs. Lors de la recherche du vecteur de gains optimal, l'exploration du dictionnaire est limitée au sous-ensemble associé à la sous-section à laquelle appartient le gain du codeur NB-AMR.

30

**[0016]** Dans "Transcodage intelligent à faible complexité entre les codeurs UIT-T G.729 (8kbit/s) et 3GPP NB-AMR (12.2 kbit/s), CORESA 2004, on propose une méthode originale de conversion pour l'excitation fixe.

**[0017]** A la connaissance des inventeurs, il n'existe aujourd'hui aucune technique de transcodage entre deux modélisations LTP multitap. Comme on l'a vu ci-avant, la plupart des solutions actuelles ne concernent que les modèles LTP monotap. Certaines techniques proposent un transcodage entre un modèle multitap et un modèle monotap mais se limitent à réduire la complexité de la recherche du retard en boucle ouverte du second format.

**[0018]** Parmi les quelques approches proposées pour réduire la complexité de la boucle fermée, certaines sont basées sur une approximation d'un filtre LTP multitap par un filtre LTP monotap (fractionnaire ou non). Par exemple, dans le cas d'une approximation d'un filtre multitap :

50

$$P_{multi}(z) = \sum_{i=-k}^k \beta_i z^{-T-i}$$

par un filtre monotap non fractionnaire  $P_{mono}(z) = \beta z^{-(T-\delta)}$ ,

on estime un gain  $\beta$  et une gigue  $\delta$  de retard tels que :

55

$$P_{mono}(z) \cong P_{multi}(z), \text{ pour tous les retards entiers } T \text{ considérés.}$$

**[0019]** L'approximation d'un modèle LTP multitap par un modèle LTP monotap est déjà exploitée depuis la norme UIT-T G.723.1, en fait pour estimer le pré-filtre adaptatif et aussi pour contrôler l'instabilité du filtre LTP. Les études menées pendant la conception du codeur selon la norme G.723.1 ont montré qu'il n'est pas toujours possible d'approximer de manière satisfaisante un filtre LTP multitap par un filtre LTP monotap, sur une large plage de retards, avec un même gain  $\beta$  et une même gigue  $\delta$  sur le retard. Pour un même vecteur de gains ( $\beta$ ), l'estimation du couple optimal ( $\beta$ ,  $\delta$ ) peut varier fortement en fonction du retard T. Dans le codeur selon G.723.1, cette difficulté a pu être contournée car la procédure de contrôle de stabilité retient le gain maximum parmi les gains estimés (qui peuvent être alors très dissemblables) et le pré-filtre adaptatif est inhibé pour tout vecteur de gains du modèle multitap lorsque, sur la plage de retards considérée, les gains estimés sont trop différents ou les giges sur le retard sont trop dissemblables ou trop grandes. Si, pour les modules de pré-filtrage adaptatif et de contrôle d'instabilité du filtre de prédiction à long terme, il est possible de contourner la difficulté d'estimation sans dégrader les performances, ces avantages sont plus difficilement atteignables avec le module d'analyse LTP lui-même qui joue un rôle crucial sur la qualité. Ainsi, selon le vecteur de gains et/ou le retard considérés, les 170 gains globaux calculés pour chaque vecteur des 170 entrées du dictionnaire, comme vu dans l'art antérieur ci-avant [1], peuvent être très éloignés des gains optimaux. De même, selon le vecteur de gains ( $\beta$ ) et/ou le retard  $\lambda$ , le calcul du retard fractionnaire  $\lambda'$ , comme vu dans l'art antérieur WO-03058407 ci-avant, peut conduire à une mauvaise détermination du retard fractionnaire.

**[0020]** Que l'approche soit analytique ou statistique, l'approximation, sur une large gamme de retards, d'un filtre LTP multitap par un seul filtre LTP monotap (ou l'approximation inverse) est trop imprécise. Pour résoudre ce problème, on pourrait, pour tenir compte de la variation du gain  $\beta$  et/ou du la gigue  $\delta$  selon le retard T, stocker pour chaque retard T un couple ( $\beta$ ,  $\delta$ ). Cependant, cette solution serait trop coûteuse en stockage puisqu'elle nécessiterait, pour chaque vecteur de gain et pour chaque retard de la plage, de stocker un couple. Dans le cas de l'approximation des filtres LTP multitap du codeur G.723.1 qui comporte deux dictionnaires multitap de 170 et 85 vecteurs, avec une plage de 125 retards, il faudrait stocker 31875 ( $=125*(170+85)$ ) couples. De plus, cette solution ne résoudrait pas les cas où l'approximation d'un multitap par un monotap est vraiment trop imprécise, voire erronée. On notera qu'à l'inverse, plusieurs couples ( $\beta$ ,  $\delta$ ) peuvent aussi constituer de bonnes approximations d'un filtre LTP multitap.

**[0021]** La présente invention vient améliorer la situation.

**[0022]** Tout d'abord, la présente invention vise le passage d'un modèle LTP à un seul coefficient (monotap) vers un modèle LTP à plusieurs coefficients (multitap) et inversement, ainsi que le passage entre deux modèles LTP multitap. En particulier, elle propose un procédé dont la complexité peut être ajustée, notamment en fonction d'un compromis désiré entre une complexité visée et une qualité souhaitée. Un dispositif pour la mise en oeuvre du procédé selon l'invention est, en outre, très utile pour les codages multiples en cascade (transcodages) ou en parallèle (multi-codages et codages multi-mode).

**[0023]** Ainsi, l'invention vise d'abord un procédé de codage tel que défini dans la revendication 1, selon un second format, à partir d'informations obtenues par la mise en oeuvre d'au moins une étape de codage selon un premier format. Les premier et second formats mettent en oeuvre, notamment pour le codage d'un signal de parole, une étape de recherche de paramètres de prédiction à long terme LTP par exploration d'au moins un dictionnaire comportant des paramètres candidats, l'un au moins des premier et second format de codage utilisant un filtrage à plusieurs coefficients (dit "*multitap*" ci-avant), pour une recherche fine des paramètres LTP.

**[0024]** Selon une définition générale de l'invention, le procédé comporte les étapes suivantes :

- a) définir des ordres d'au moins un dictionnaire qu'utilise le second format de codage,
- b) récupérer une information a priori, obtenue suite à la détermination des paramètres LTP au cours du codage selon le premier format, pour sélectionner au moins un ordre dudit dictionnaire,
- c) appliquer l'ordre sélectionné aux candidats dudit dictionnaire pour choisir un nombre limité de premiers candidats, et
- d) pour effectuer le second codage, mener la recherche LTP uniquement parmi ledit nombre limité de candidats.

**[0025]** L'invention se distingue donc des solutions existantes par la définition d'ordres dans le dictionnaire et l'exploitation de ces ordres dans la procédure d'exploration du dictionnaire.

**[0026]** D'autres caractéristiques et avantages de l'invention apparaîtront à l'examen de la description détaillée ci-après, et des dessins annexés sur lesquels :

- la figure 1a représente schématiquement un système de transcodage intelligent utilisant un dispositif de codage selon le second format au sens de l'invention,
- la figure 1b représente schématiquement un système de codage multiple en parallèle, utilisant un dispositif de

- codage selon le second format au sens de l'invention,
- la figure 2 illustre les étapes principales du procédé au sens de l'invention,
  - la figure 3 représente schématiquement les moyens mis en oeuvre par un dispositif de codage au sens de l'invention,
  - la figure 4a représentant un schéma de principe d'un codeur CELP (pour "Code Excited Linear Prediction"),
  - 5 - la figure 4b représente schématiquement les étapes de l'analyse LTP d'un codeur selon la norme UIT-T G.729,
  - la figure 4c représente schématiquement les étapes de l'analyse LTP d'un codeur selon la norme UIT-T G.723.1 (6,3 kbit/s),
  - la figure 5a illustre une correspondance entre les trames d'un codeur selon la norme UIT-T G.723.1 (30 ms) et les trames d'un codeur selon la norme UIT-T G.729 (10 ms),
  - 10 - la figure 5b illustre une correspondance entre les sous-trames du codeur G.729 (5 ms) et les sous-trames du codeur G.723.1 (7,5 ms),
  - la figure 6 illustre la recherche du pitch en boucle ouverte du G.729 à partir des valeurs de pitch du G.723.1,
  - les figures 7a et 7b illustrent respectivement l'association entre sous-trames paires (respectivement impaires) du codeur G.729 et le jeu de paramètres LTP issu du codeur G.723.1 en tant que codeur selon le premier format,
  - 15 - la figure 8 représente un tableau d'association des sous-trames du G.723.1 (colonne de droite CD) aux sous-trames du G.729 (colonne de gauche CG),
  - les figures 9a et 9b représentent des histogrammes de tailles d'exploration réduites (nombre d'occurrences en ordonnée) dans des dictionnaires (initialement de 85 vecteurs pour la figure 9a et de 170 vecteurs pour la figure 9b), et garantissant moins de 1% de réduction de qualité selon le critère CELP, et
  - 20 - la figure 10 représente schématiquement la sélection de N éléments du second dictionnaire quand plusieurs ordres sont constitués, dans une réalisation particulière.

**[0027]** La présente invention s'inscrit donc dans le cadre du codage multiple en cascade ou en parallèle ou dans tout autre système utilisant, pour représenter la périodicité à long terme d'un signal, une modélisation de type monotap ou multitap. L'invention permet à partir de la connaissance des paramètres d'un premier modèle de déterminer les paramètres d'un second modèle dans le cas où au moins l'un des deux modèles utilise une modélisation multitap. Par souci de concision, seul le cas d'un passage d'un premier modèle vers un second est décrit mais on comprendra que l'invention s'applique aussi dans le cas de passage de m ( $m \geq 1$ ) premiers modèles vers n ( $n \geq 2$ ) seconds modèles (où m et n sont absolument quelconques).

**[0028]** En référence aux figures 1a et 1b, on considère donc le cas de deux modélisations LTP d'un signal correspondant à deux systèmes de codage COD1 et COD2. Il peut s'agir d'un passage du premier système de codage COD1 au second système de codage COD2, en cascade notamment par transcodage intelligent (figure 1a), ou en parallèle notamment en optimisant le codage multiple (figure 1b). Le premier codeur a effectué son opération de codage sur un signal donné (par exemple le signal original  $s_0$ ). On dispose donc des paramètres LTP, notés LTP1, choisis par le premier codeur COD1. Ce codeur a déterminé ces paramètres par une technique qui lui est propre lors du processus de codage. Le second codeur COD2 doit réaliser lui aussi son codage. Dans le cas du transcodage, le second codeur COD2 ne dispose que du train binaire BS1 généré par le premier codeur COD1 et incluant alors les codes binaires des paramètres LTP1. L'invention est donc ici applicable au transcodage intelligent. Dans le cas du codage multiple en parallèle, le second codeur COD2 dispose aussi du signal d'origine  $s_0$  (ou d'une version dérivée) dont dispose le premier codeur COD1 et l'invention s'applique ici au multi-codage intelligent. On indique que l'invention peut aussi s'appliquer au cas particulier du codage multiple en parallèle qu'est le codage multi-mode à décision a posteriori.

**[0029]** La présente invention porte sur la détermination d'un paramètre d'un modèle LTP, noté LTP2, à partir d'au moins un paramètre LTP1 d'un autre modèle LTP, quand l'un au moins des deux modèles est un modèle multitap. Au lieu de rechercher le paramètre du second format de codage dans son ensemble de définition (ou "dictionnaire"), l'invention prévoit les étapes suivantes, en se référant maintenant à la figure 2 :

- dans le dictionnaire DIC2 du second format de codage (portant la référence 25 sur la figure 2), on détermine initialement des ordres ORD1, ORD2, ..., ORDn (étape 25b de la figure 2),
- 50 - à partir de l'un au moins des paramètres LTP1 du premier format de codage, on sélectionne au moins un ordre ORD(DIC2) du dictionnaire du second format (étape 26),
- à l'étape 27, on obtient une succession ordonnée des éléments du dictionnaire  $e_i^2, e_j^2, e_k^2, \dots$ ,
- 55 - on limite avantageusement l'exploration aux premiers éléments  $e_i^2, e_j^2$  du dictionnaire DIC2 ainsi ordonné (étape 29), ce nombre d'éléments étant de préférence choisi selon le compromis qualité/complexité voulu (qualité visée/complexité autorisée), à une étape 28.

**[0030]** Ainsi, on comprendra que l'on peut limiter, par la mise en oeuvre de l'invention, le nombre d'éléments du second dictionnaire DIC2 sur lequel la recherche LTP va porter pendant le second codage COD2, tout en assurant une bonne qualité du codage COD2. Sur la figure 2, on a séparé en deux blocs 20 et 24 les opérations menées respectivement par le premier codeur COD1 et le second codeur COD2, ce dernier disposant du dictionnaire DIC2 (référence 25). De son côté, le premier codeur COD1 a déterminé les paramètres LTP1, à l'étape 21, en utilisant au moins son dictionnaire DIC1 (étape 22). On comprendra alors que la manière dont le premier codeur COD1 a déterminé ses paramètres LTP1, typiquement à partir du signal d'origine  $s_0$ , constitue une information *a priori* (étape 23) qui peut être utilisée par le second codeur pour ordonner son dictionnaire DIC2. Finalement, les paramètres LTP2 obtenus (étape 30) en appliquant la classification du dictionnaire du second codeur au sens de l'invention, pourront eux-mêmes servir à la classification d'un dictionnaire selon encore un troisième format de codage (non représenté), le cas échéant, et ainsi de suite pour un transcodage en cascade ou un codage multiple en parallèle.

**[0031]** On notera que la figure 2 n'est donnée ici qu'à des fins principalement didactiques. Par exemple, la notation  $e_i^2$ ,  $e_j^2$ ,  $e_k^2$ , ... des éléments du dictionnaire DIC2 n'est pas réellement conventionnelle, comme on le verra plus loin. Par ailleurs, la classification du dictionnaire DIC2 (étape 25b) et la limitation de ses éléments à prendre en compte pour la recherche en fonction du critère qualité/complexité (étape 28) peuvent être menées conjointement sensiblement en une même étape. Enfin, on a représenté sur la figure 2 un premier codeur COD1 délivrant les informations *a priori* (étape 23) au second codeur COD2. Néanmoins, en variante, le second codeur COD2 peut simplement récupérer du premier codeur COD1 les codes binaires des paramètres LTP1 que le premier codeur a déterminé et retrouver ces informations *a priori* grâce notamment à la connaissance du type de codage et du dictionnaire utilisé par le premier codeur COD1.

**[0032]** On a représenté sur la figure 3 un dispositif de codage selon le second format, au sens de l'invention. Ce dispositif est agencé pour utiliser des informations de codage par la mise en oeuvre d'un codage selon un premier format (ici les paramètres LTP1 récupérés du codage selon le premier format COD1). Le dispositif au sens de l'invention comporte, dans l'exemple représenté :

- une mémoire MEM stockant une table de correspondance définissant, en fonction des paramètres LTP1 déterminés par le premier format de codage, des ordres d'un dictionnaire qu'utilise le second format de codage,
- des moyens, tels qu'une interface 31, pour récupérer un signal donnant au moins une information *a priori* sur des paramètres LTP1 au cours d'un codage selon le premier format,
- des moyens 32 actifs sur réception de ce signal pour consulter la table de correspondance et sélectionner au moins un ordre du dictionnaire du second format,
- des moyens de calcul, tel qu'un processeur 35, pour :
  - ordonner le dictionnaire 33 du second format selon l'ordre sélectionné, en vue de choisir un nombre limité de premiers candidats dans le dictionnaire 33, et
  - poursuivre le codage selon le second format, avec d'autres modules 34 le cas échéant, en menant la recherche LTP uniquement parmi ce nombre limité de candidats.

**[0033]** Bien entendu, le processeur 35 gère tout ou partie des modules du dispositif. A cet effet, il peut être animé par un produit programme d'ordinateur. La présente invention vise d'ailleurs un tel produit programme d'ordinateur, stocké dans une mémoire d'une unité de traitement ou sur un support amovible destiné à coopérer avec un lecteur de ladite unité de traitement ou téléchargeable d'un site distant, et comportant des instructions pour la mise en oeuvre de tout ou partie des étapes du procédé selon l'invention.

**[0034]** On comprendra en particulier que le dispositif COD2, au sens de l'invention, peut directement récupérer les paramètres LTP1 du premier codeur COD1 pour en déduire l'information *a priori* précitée et, de là, l'ordre de son dictionnaire DIC2, ou, en variante recevoir du premier codeur COD1 directement les informations *a priori* sur l'ordre de son dictionnaire, du premier codeur COD1. Dans ce dernier cas, le premier codeur COD1 joue déjà un rôle particulier dans l'invention.

**[0035]** La présente invention vise aussi un système incluant le premier codeur et le dispositif au sens de l'invention. En effet, le dispositif de la figure 3 peut être inséré dans un système de codage mettant en oeuvre au moins un premier et un second format de codage. Ce système comporte alors au moins un dispositif de codage selon le premier format COD1 et un dispositif de codage au sens de l'invention et appliquant alors second format COD2. A ce titre, l'invention vise un tel système. Le dispositif de codage selon le premier format et le dispositif de codage selon le second format peuvent être mis en cascade, pour un transcodage, comme représenté sur la figure 1a. En variante, le dispositif de codage selon le premier format et le dispositif de codage selon le second format peuvent être mis en parallèle, pour un codage multiple, comme représenté sur la figure 1 b.

**[0036]** Dans la mise en oeuvre de l'invention, on retiendra que le second codeur COD2 peut récupérer du premier codeur COD1 (lorsque ce dernier a déterminé les paramètres LTP1) des informations qui vont lui permettre d'ordonner son dictionnaire DIC2 (voir figure 2). Ensuite, une recherche LTP uniquement parmi les premiers éléments ( $e_1^2, e_j^2$ ) du dictionnaire DIC2 ainsi ordonné permettra de conserver une bonne qualité du second codage.

**[0037]** Avantageusement, l'exploitation des ordres du second dictionnaire DIC2 offre une grande flexibilité sur le nombre d'éléments ordonnés à explorer. Il est alors possible :

- d'ajuster librement le compromis qualité/complexité,
- ou encore, pour une complexité donnée, optimiser la qualité,
- ou, inversement, minimiser la complexité pour une qualité donnée.

**[0038]** Cet ajustement peut être effectué au début du traitement. Il peut aussi être effectué à chaque bloc à traiter en fonction de paramètres du premier format de codage et/ou des caractéristiques du signal à coder (par exemple, en fonction d'un critère de voisement). Pour un même bloc, la complexité peut aussi varier en fonction des sous-trames LTP. L'invention offre une grande flexibilité qui permet de répartir dynamiquement la puissance de calcul disponible entre les modules du second codeur et/ou les ressources pour traiter les sous-trames LTP.

**[0039]** Préférentiellement, c'est à partir d'une partition initiale du dictionnaire DIC1 associé à un paramètre du premier modèle LTP que l'on détermine des ordres du dictionnaire DIC2 associé à un paramètre du second modèle LTP. On indique que la détermination d'un ordre consiste à classer les éléments du second dictionnaire DIC2 selon un certain critère. Un classement (ou "ordre") est donné par une indexation des éléments du dictionnaire DIC2.

**[0040]** Plusieurs types de partition du premier dictionnaire DIC1 peuvent être prévus. Un premier exemple est la partition élémentaire d'un dictionnaire DIC1 de N éléments en N classes disjointes de taille 1. N ordres du deuxième dictionnaire sont alors déterminés. Des partitions plus élaborées peuvent être choisies, notamment par des techniques connues en soi de quantification (vectorielle ou scalaire) ou de classification des données.

**[0041]** Avantageusement, il est possible de regrouper des ordres similaires, ce qui revient à modifier la partition initiale du premier dictionnaire et, en conséquence, le nombre d'ordres du deuxième dictionnaire. On peut aussi recalculer les ordres une fois qu'ils ont été regroupés. Les procédures de détermination de la partition du premier dictionnaire en N classes et de calcul des N ordres du second dictionnaire peuvent être itérées, le nombre N pouvant d'ailleurs varier au cours des itérations. En variante ou en complément, pour limiter la mémoire nécessaire au stockage des ordres du second dictionnaire, pour chacun de ces ordres, un nombre maximal d'éléments à retenir est choisi, ce nombre pouvant différer selon les ordres et/ou les classes du premier dictionnaire.

**[0042]** Dans une variante encore, les classes du premier dictionnaire ne sont pas forcément disjointes. Typiquement, un même élément peut être associé à plus d'un ordre du second dictionnaire. Le choix de l'ordre ou la combinaison des ordres peut alors prendre en compte d'autres facteurs que le paramètre LTP courant du premier dictionnaire.

**[0043]** Initialement, le nombre d'ordres et les ordres qui conviennent dans le second dictionnaire sont déterminés par une étude statistique et/ou analytique, en fonction de jeux successifs de paramètres LTP selon le premier modèle. Cette étude définit donc, pour chaque classe de la partition du dictionnaire associé à un paramètre LTP du premier format, un classement du dictionnaire d'un paramètre du second format. Une étude statistique a été réalisée sur un banc hors ligne en associant dans un même codeur le modèle LTP du premier format et le modèle LTP du deuxième format. La mise en parallèle des deux analyses LTP a été la configuration préférée d'apprentissage. Bien entendu, d'autres configurations peuvent être utilisées, notamment un tandem conventionnel qui met en cascade les deux codages. L'étude statistique assure, pour chaque élément du premier dictionnaire (ou chaque classe de sa partition), un classement des éléments du second dictionnaire selon un certain critère. Préférentiellement, ce critère évalue l'impact sur la qualité du signal restitué. En effet, le critère de qualité peut être celui utilisé au codage pour sélectionner le second paramètre LTP. Bien entendu, d'autres critères peuvent être utilisés, notamment la sollicitation d'un élément du second dictionnaire pour une classe du premier dictionnaire. En outre, une combinaison de critères peut aussi être utilisée.

**[0044]** Une étude analytique peut aussi être effectuée pour déterminer des ordres du second dictionnaire en fonction d'une partition du premier dictionnaire. Préférentiellement, l'étude analytique vient en complément de l'étude statistique décrite ci-avant. Elle est préférentiellement limitée aux parties de dictionnaire qui conduisent à des approximations analytiques satisfaisantes.

**[0045]** On décrit maintenant la détermination d'un paramètre LTP du second format de codage, à partir des paramètres LTP selon le premier format de codage.

**[0046]** Dans le cadre de la conception d'algorithmes pour une exploration restreinte du second dictionnaire connaissant les paramètres LTP choisis par le premier format de codage, on exploite préférentiellement la partition d'un premier dictionnaire et les ordres du second dictionnaire qui sont associés à cette partition du premier dictionnaire.

**[0047]** Par souci de clarté de l'exposé, on décrit d'abord les principes de l'algorithme utilisé quand les deux formats



de codage ont des sous-trames LTP de durée identique. A chaque sous-trame courante du deuxième format de codage correspond une seule sous-trame du premier format de codage. Pour cette première sous-trame, le premier format de codage a sélectionné un jeu de paramètres LTP (dit "*premier jeu LTP1*"). Grâce à la partition du dictionnaire associé à l'un des paramètres LTP du premier modèle, on sélectionne un ordre d'exploration du second dictionnaire en choisissant l'ordre associé à la classe de l'élément du premier jeu LTP1. Puis, le second dictionnaire est exploré suivant l'ordre ainsi déterminé. De plus, en fonction d'un compromis qualité/complexité et/ou éventuellement du nombre maximal d'éléments du second dictionnaire retenu pour la classe, le nombre d'éléments testés est restreint. De manière générale, on retiendra donc que, parmi tous les éléments du second dictionnaire, seuls sont testés les premiers éléments déterminés par l'ordre qui a été choisi.

**[0048]** Quand les deux formats de codage ont des sous-trames LTP de durées différentes, il advient qu'une sous-trame courante du deuxième format peut correspondre à plus d'une sous-trame du premier format. Cette situation est illustrée sur la figure 5b, à titre d'exemple. Pour ces premières sous-trames, le premier format de codage a sélectionné des jeux de paramètres LTP. Grâce à la partition du dictionnaire associé à l'un des paramètres LTP du premier modèle, on présélectionne des ordres d'exploration du second dictionnaire en choisissant les ordres associés aux classes des éléments des premiers jeux. Il se peut qu'un seul ordre soit finalement sélectionné si les paramètres choisis pour les premières sous-trames appartiennent à la même classe de la partition du premier dictionnaire. Il s'agit toutefois d'un cas particulier. On est alors ramené au schéma précédent correspondant à des sous-trames LTP de durée identique. Si, au contraire, plus d'un ordre a été présélectionné, on peut ne retenir qu'un seul ordre (par exemple l'ordre le plus présélectionné), ou encore celui qui correspond à la sous-trame du premier format qui couvre le plus la sous-trame courante du deuxième format.

**[0049]** Selon le type de paramètre LTP de la partition du premier dictionnaire, d'autres critères peuvent être retenus. Au lieu de ne retenir qu'un seul ordre, une autre solution consiste à combiner au moins une partie des différents ordres présélectionnés. Plusieurs procédures de combinaison sont possibles. Par exemple, si K ordres ont été retenus, on examine d'abord le premier élément de chacun des K ordres, en éliminant les redondances éventuelles. On obtient  $K_1$  éléments ( $K_1 \leq K$ ). Puis, on ajoute  $K_2$  éléments, tels que  $K_2 \leq K$  et  $K_2 \leq N - K_1$ , choisis dans l'ensemble constitué par le deuxième élément des K ordres (en éliminant les redondances éventuelles), et ainsi de suite jusqu'à obtenir N éléments, N étant le nombre maximal d'éléments du second dictionnaire à tester. On a représenté schématiquement cette sélection de N éléments  $e_1, e_2, \dots, e_K, \dots$  en tant que premiers éléments de K ordres ORD1, ORD2, ..., ORDK, sur la figure 10. Le nombre N d'éléments retenus dans l'ensemble ENS peut être choisi par exemple en fonction de la complexité maximale autorisée. Dans ce classement, il est aussi possible de privilégier les éléments les plus souvent classés parmi les premiers.

**[0050]** En variante, on peut aussi constituer K sous-ensembles des classements en présélectionnant les  $N_i$  ( $\leq N$ ) premiers éléments de chaque classement  $C_i$  ( $1 \leq i \leq K$ ). Le choix des  $N_i$  est tel que  $\sum N_i \geq N$  et permet de traiter équitablement les classements ou, au contraire, de privilégier certains classements. Puis, on sélectionne tous les éléments présents dans les K sous-ensembles, ensuite les éléments présents dans K-1 sous-ensembles et ainsi de suite jusqu'à retenir N éléments. Si N éléments n'ont pas été ainsi obtenus, on complète le nombre d'éléments en prenant par exemple successivement les éléments suivants dans les K sous-ensembles.

**[0051]** On peut évidemment combiner certaines de ces stratégies de classement. On indique de manière générale que le second dictionnaire est préférentiellement exploré suivant un ordre "*dynamique*" ainsi déterminé. Cette procédure de constitution d'ordre dynamique à partir d'ordres prédéterminés et stockés peut aussi être appliquée quand les classes de la partition ne sont pas disjointes et qu'un élément du premier dictionnaire appartient à plus d'une classe.

**[0052]** On décrit ci-après trois cas de passage d'un premier modèle LTP vers un deuxième modèle LTP, illustrant l'application de l'invention à différents modèles et types de paramètres LTP. Bien entendu, bien que les exemples ne soient donnés que pour un premier et un second dictionnaire, l'invention se généralise aisément à plus d'un premier et/ou second dictionnaire.

#### \* Cas du passage d'un modèle monotap vers un modèle multitap

**[0053]** On dispose des paramètres du modèle monotap d'un format COD1 et l'on cherche à déterminer à moindre coût de calcul et/ou de ressources ceux du modèle multitap d'un format COD2. Pour chaque sous-trame, le codeur COD1 a déterminé le couple  $(\lambda_e, \beta_e)$  de paramètres du filtre LTP monotap. Le codage d'une sous-trame de COD2 requiert la détermination de couples  $(\lambda_s, (\beta_i)_s)$  (où i est un indice de gain) de paramètres du filtre LTP multitap. Le jeu de paramètres du premier modèle est donc  $(\lambda_e, \beta_e)$ . Le jeu de paramètres du deuxième modèle est  $(\lambda_s, (\beta_i)_s)$ .

**[0054]** La détermination du retard  $\lambda_s$  est faite par une des méthodes connues de l'état de l'art. Par exemple, il est possible d'utiliser la méthode de transcodage intelligent qui détermine directement ce retard  $\lambda_s$  en choisissant comme retard, celui déterminé par COD1 sur sa sous-trame qui partage le plus d'échantillons avec la sous-trame courante de COD2 (si ce retard  $\lambda_e$  est fractionnaire, on prend sa partie entière ou l'entier le plus proche). Cette situation sera décrite plus loin en référence notamment aux figures 7a et 7b.

**[0055]** On détermine alors, avec une complexité faible au sens de l'invention, le vecteur de gains  $(\beta_i)_s$  pour chaque

sous-trame de COD2 à partir de l'un au moins des gains  $\beta_e$  des sous-frames de COD1. Par une étude associant les deux modèles LTP, on a effectué une partition du premier dictionnaire (ici le dictionnaire des gains scalaires  $\beta_e$ ). On détermine ensuite des ordres du deuxième dictionnaire, associés à cette partition. Ces ordres correspondent ici à l'ensemble des vecteurs de gains  $(\beta_i)_s$ . A partir des gains LTP scalaires  $\beta_e$  choisis par le premier format COD1 pour ses sous-frames correspondant à une sous-trame courante de COD2, les ordres du deuxième dictionnaire associés aux classes de ces gains scalaires sont présélectionnés. Puis, un seul de ces ordres peut être retenu, ou encore, un ordre est dynamiquement constitué. Enfin, les N premiers vecteurs de gains déterminés par cet ordre sont testés pour sélectionner le meilleur vecteur (selon un critère tel que le critère habituel CELP). On rappelle que, grâce aux ordres, le nombre N peut être aisément ajustable en fonction par exemple du compromis qualité/complexité désiré. En général, N est bien inférieur à la taille du second dictionnaire.

**[0056]** Selon l'un des avantages de la présente invention, on détermine ainsi le vecteur de gains optimal d'un filtre LTP multitap d'un second format de codage à partir d'au moins un gain d'un filtre LTP monotap d'un premier format, en réduisant considérablement la complexité d'exploration du second dictionnaire des vecteurs de gains et en limitant le nombre de vecteurs de gains à tester. Contrairement à la référence [2] donnée ci-avant où, pour chaque gain monotap, un sous-ensemble de vecteurs de gains de taille fixe est associé, la solution au sens de l'invention permet d'ajuster l'exploration du dictionnaire en fonction de la qualité visée et des contraintes de complexité. On comprendra que l'invention fait intervenir davantage les différents ordres du dictionnaire de vecteurs de gains que des sous-ensembles prédéfinis et fixes comme dans la référence précitée.

**[0057]** Dans le cas d'un transcodage intelligent du codeur UIT-T G.729 à 8 kbit/s vers le codeur UIT-T G.723.1 à 6,3 kbit/s, qui sera décrit plus loin en tant qu'exemple de réalisation, les étapes exposées ci-avant pourront s'appliquer à la focalisation de la recherche en boucle fermée dans les deux dictionnaires de vecteurs de gains du G.723.1 à partir des gains LTP du codeur G.729.

#### \* Cas du passage d'un modèle multitap vers un modèle monotap

**[0058]** Ce cas de figure est l'inverse du précédent. On dispose des paramètres du modèle LTP multitap d'un premier format COD1 et l'on cherche à déterminer à moindre coût ceux du modèle LTP monotap d'un second format COD2. Le jeu de paramètres du premier modèle s'écrit donc  $(\lambda_e, (\beta_i)_e)$  (où i est un indice de gain), tandis que le jeu de paramètres du deuxième modèle s'écrit  $(\lambda_s, \beta_s)$ . A partir d'au moins un jeu de paramètres sélectionnés par le premier codeur COD1, on cherche à obtenir un retard  $\lambda_s$  et un gain  $\beta_s$  pour le format COD2. Par une étude associant les deux modèles LTP, on a effectué une partition du premier dictionnaire qui est, dans ce cas, celui des vecteurs de gains  $(\beta_i)_e$ . On détermine ensuite, au sens de l'invention, des ordres du deuxième dictionnaire, associés à la partition du premier dictionnaire. Ici, le deuxième dictionnaire est constitué de l'ensemble des valeurs de gigue  $(\lambda_e - \lambda_s)$ . A partir des vecteurs de gains  $(\beta_i)_e$  choisis par le premier format COD1, pour ses sous-frames qui correspondent à la sous-trame courante de COD2, les ordres du deuxième dictionnaire associés aux classes de ces vecteurs de gains sont présélectionnés. Ensuite, un seul de ces ordres peut être retenu, ou encore, un ordre peut être dynamiquement constitué. Enfin, les valeurs de "voisinage" ainsi déterminées autour d'un ou plusieurs retards d'ancrage  $\lambda'_s$  sont explorées. La détermination du (ou des) retards d'ancrage est faite par une méthode connue dans l'état de l'art.

**[0059]** La présente invention propose donc une solution originale permettant de réduire la complexité de la détermination du retard  $\lambda_s$ , en réduisant le nombre de valeurs de retards testés d'un modèle LTP monotap d'un second format de codage à partir de la connaissance des paramètres d'un modèle LTP multitap d'un premier format de codage. La plupart des méthodes de l'art antérieur n'utilisent que le retard sans exploiter le vecteur de gains. Comme dans le document WO-03058407, on utilise ici les deux types de paramètres. Néanmoins, contrairement à l'enseignement de cette dernière référence, un vecteur de gains pointe sur un ensemble de plusieurs valeurs de gigue et non pas sur une seule valeur comme dans cette référence. Selon l'un des avantages que procure l'invention, on s'affranchit ainsi des problèmes liés à l'approximation d'un filtre LTP multitap par un seul filtre monotap.

**[0060]** Dans une variante avantageuse, pour limiter le stockage, les voisinages ordonnés sont des intervalles de taille croissante. Cette mesure est particulièrement avantageuse pour focaliser la recherche en boucle ouverte et/ou fermée. Un exemple de réalisation sera décrit plus loin, relatif à la recherche en boucle fermée du retard LTP du codeur UIT-T G.729 à 8 kbit/s à partir des paramètres LTP du codeur UIT-T G.723.1 à 6,3 kbit/s.

#### \* Cas du passage d'un modèle multitap vers un modèle multitap

**[0061]** A la connaissance des inventeurs, ce cas n'a jamais été étudié dans l'art antérieur.

**[0062]** On dispose des paramètres du modèle multitap d'un premier format COD1 et l'on cherche à déterminer à moindre coût ceux du modèle multitap d'un second format COD2. Le jeu de paramètres du premier modèle s'écrit donc  $(\lambda_e, (\beta_i)_e)$ . Le jeu de paramètres du deuxième modèle s'écrit aussi  $(\lambda_s, (\beta_i)_s)$ . A partir d'au moins un jeu de paramètres sélectionné par le premier format COD1, on cherche à obtenir un retard  $\lambda_s$  et un vecteur de gains  $(\beta_i)_s$  pour le second

format COD2.

[0063] La détermination du retard  $\lambda_s$  à partir d'au moins un retard  $\lambda_e$  est faite par une méthode connue de l'état de l'art. On retiendra que la mise en oeuvre de la présente invention permet ici de déterminer avec une complexité faible le vecteur de gains  $(\beta_i)_s$  pour chaque sous-trame du second format COD2 à partir d'au moins un vecteur de gains  $(\beta_i)_e$  des sous-frames du premier format COD1. Par une étude associant les deux modèles LTP multitap, on a effectué, au sens de l'invention, une partition du premier dictionnaire qui est dans ce cas celui des vecteurs de gains  $(\beta_i)_e$ . On détermine ensuite les ordres du deuxième dictionnaire (ici celui des vecteurs de gains  $(\beta_i)_s$ ) qui sont associés à cette partition. A partir des vecteurs de gains  $(\beta_i)_e$  choisis par le premier format COD1 pour ses sous-frames qui correspondent à la sous-trame courante du second format COD2, les ordres du deuxième dictionnaire associés aux classes de ces vecteurs de gains sont présélectionnés. Ensuite, un seul de ces ordres peut être retenu, ou encore un ordre peut être dynamiquement et évolutivement constitué. Enfin, les premiers vecteurs de gains déterminés par cet ordre sont testés pour sélectionner le meilleur.

[0064] Un exemple de réalisation entre les débits 6,3 kbit/s et 5,3 kbit/s du codeur UIT-T G.723.1 illustrant ce dernier cas est présenté plus loin.

## \*\* EXEMPLES DE REALISATION

[0065] On présente ci-après trois exemples de réalisation qui visent le transcodage entre deux formats de codage différents UIT-T G.729 et UIT-T G.723.1 pour les deux premiers, et un changement de débit au sein d'un codeur multidébit (UIT-T G.723.1) pour le dernier. Une description de ces deux codeurs UIT-T est d'abord donnée ainsi que leurs modélisations LTP.

### **Codeurs UIT-T G. 729 à 8 kbit/s et UIT-T G. 723.1 (6,3 kbit/s et 5,3 kbit/s)**

[0066] Ces deux codeurs appartiennent à la famille des codeurs CELP, codeurs à analyse par synthèse.

#### **Codeurs à analyse par synthèse**

[0067] Dans ces codeurs, le modèle de synthèse est utilisé pour extraire les paramètres modélisant les signaux à coder. Ces signaux peuvent être échantillonnés à la fréquence téléphonique ( $F_e = 8$  kHz) ou une fréquence plus élevée, par exemple à 16 kHz pour le codage en bande élargie (bande passante de 50 Hz à 7 kHz). Selon l'application et la qualité désirée, le taux de compression varie de 1 à 16 de sorte que ces codeurs fonctionnent à des débits de 2 à 16 kbit/s en bande téléphonique, et à des débits de 6 à 32 kbit/s en bande élargie. Le dispositif de codage et décodage numérique de type CELP, codeur à analyse par synthèse le plus utilisé actuellement pour le codage des signaux de parole, est présenté sur la 4a. Le signal de parole  $s_0$  est échantillonné et converti en une suite de blocs de  $(L')$  échantillons appelés trames. En général, chaque trame est découpée en blocs plus petits de  $(L)$  échantillons, appelés sous-frames. Chaque bloc est synthétisé en filtrant une forme d'onde extraite d'un répertoire (appelé aussi dictionnaire d'excitation fixe), multipliée par un gain, à travers deux filtres variant dans le temps. Le dictionnaire d'excitation est un ensemble fini de formes d'ondes de  $L$  échantillons. Le premier filtre est le filtre de prédiction à long terme. Une analyse "LTP" (Long Term Prédiction) permet d'évaluer les paramètres de ce prédicteur à long terme qui exploite la périodicité des sons voisés. Ce prédicteur est équivalent à un dictionnaire stockant l'excitation passée pour différents retards. Ce dictionnaire est généralement appelé "dictionnaire d'excitation adaptative". Le second filtre est le filtre de prédiction à court terme. Les méthodes d'analyse "LPC" (Linear Prediction Coding) permettent d'obtenir ces paramètres de prédiction à court terme, représentatifs de la fonction de transfert du conduit vocal et caractéristiques du spectre du signal.

[0068] Ainsi, en se référant à la figure 4a représentant un schéma de principe d'un codeur CELP, le signal de parole  $s_0$  subit l'analyse LPC 41 (non représenté en détail), ainsi qu'une analyse LTP avec une construction du répertoire des excitations fixes 46 et des excitations adaptatives 45 pour alimenter le filtre de synthèse 44. On prévoit en outre dans la boucle ainsi constituée un module de pondération perceptuelle 42 et un module de minimisation de l'erreur 43.

[0069] Le procédé utilisé pour déterminer la séquence d'innovation est donc la méthode d'analyse par synthèse. Au codeur, un grand nombre de séquences d'innovation du dictionnaire d'excitation sont filtrées par les deux filtres LTP et LPC, et la forme d'onde sélectionnée est celle produisant le signal synthétique le plus proche du signal d'origine selon un critère de pondération perceptuelle, connu en général sous le nom de critère CELP.

#### **Modèle LTP du G.729 à 8 kbit/s (monotap)**

[0070] Le codeur UIT-T G.729 opère sur un signal de parole limité en bande à 3,4 kHz, échantillonné à 8 kHz et découpé en trames de 10 ms (soit 80 échantillons par trame). Chaque trame est divisée en deux sous-frames (numérotées ci-après 0 et 1) de 40 échantillons (5 ms). Le modèle LTP du codeur UIT-T G.729 se base sur une modélisation monotap

à résolution fractionnaire. A chaque trame, l'analyse LTP détermine un retard  $\lambda_i$  et un gain  $\beta_i$  pour chaque sous-trame. La figure 4b en présente les principales étapes. A chaque trame, une recherche du retard en boucle ouverte, noté  $\lambda_{OL}$ , est effectuée dans la plage de valeurs [20 ; 143] (étape 401). Puis, le retard de la première sous-trame est recherché en boucle fermée autour du retard en boucle ouverte  $\lambda_{OL}$  sur la plage [ $\lambda_{OL}-3$ ;  $\lambda_{OL}+3$ ] (étape 402). En utilisant donc

la plage [ $19\frac{1}{3}$ ;  $84\frac{2}{3}$ ] et en résolution entière dans la plage [85; 143].

[0071] Puis, le retard  $\lambda_1$  de la seconde sous-trame est déterminé avec une résolution fractionnaire de 1/3 par analyse par synthèse autour de  $\lambda_0$  sur la plage [ $\text{int}(\lambda_0)-5_{2/3}$ ;  $\text{int}(\lambda_0)+4_{2/3}$ ],  $\text{int}(\lambda_0)$  étant la partie entière du retard éventuellement fractionnaire  $\lambda_0$  (étape 404). Pour chaque sous-trame, le gain  $\beta$  est calculé une fois le retard en boucle fermée déterminé (étapes 403 et 405). Après la recherche de l'excitation fixe, le gain  $\beta$  est quantifié conjointement avec le gain de l'excitation fixe par une quantification vectorielle sur sept bits. L'ensemble de définition (ou dictionnaire) du gain LTP monotap du G.729 a donc une taille de 128.

### Modèle LTP du G.723.1 (multitap)

[0072] Le codeur UIT-T G.723.1 opère sur un signal de parole limité en bande à 3,4 kHz, échantillonné à 8 kHz et découpé en trames de 30 ms (soit 240 échantillons par trame). Chaque trame comporte 4 sous-trames de 7,5 ms (60 échantillons) regroupées 2 par 2 dans des super sous-trames de 15 ms (120 échantillons). Le codeur UIT-T G.723.1 utilise une modélisation multitap d'ordre 5. Les coefficients du prédicteur à long terme sont quantifiés vectoriellement au moyen de deux dictionnaires préalablement stockés à 85 ou 170 entrées pour le mode à 6,3 kbit/s, tandis que le mode à 5,3 kbit/s n'utilise que le dictionnaire à 170 entrées. Dans le mode 6,3 kbit/s, le choix du dictionnaire exploré dépend de la valeur de retard des sous-trames paires.

[0073] La figure 4c illustre les principales étapes de l'analyse LTP du codeur G.723.1. A chaque trame, deux analyses LTP en boucle ouverte (une fois par super sous-trame) sont effectuées afin d'estimer un retard  $\lambda_{OL}^i$  ( $i = 0$  ou  $1$ ) sur la plage [18; 142] pour chaque bloc de 120 échantillons (étape 410). Puis, pour chaque super sous-trame, deux analyses LTP en boucle fermée (une pour chaque sous-trame) sont effectuées. Les retards  $\lambda_{2i}$  des sous-trames paires (sous-trames 0 et 2) sont recherchés en boucle fermée autour du retard  $\lambda_{OL}^i$  correspondant sur la plage [ $\lambda_{OL}^i-1$ ;  $\lambda_{OL}^i+1$ ]. Conjointement avec cette recherche, le dictionnaire de vecteurs de gains est aussi exploré par analyse par synthèse (étape 411). Pour les sous-trames impaires (sous-trames 1 et 3), une recherche similaire (recherche conjointe du vecteur de gains et du retard en boucle fermée) est effectuée et la recherche d'un retard  $\lambda_{2i+1}$  en boucle fermée est limitée au voisinage du retard en boucle fermée de la sous-trame précédente [ $\lambda_{2i-1}$ ;  $\lambda_{2i}+2$ ] (étape 412).

### **Premier exemple de réalisation : détermination des paramètres LTP multitap du G.723.1 à 6,3 kbit/s à partir des paramètres LTP Monotap du G.729 à 8 kbit/s**

[0074] Comme présenté sur la figure 5a, en prenant classiquement une origine temporelle commune, une trame de codage G.723.1 correspond à trois trames de codage G.729. Il apparaît donc que les sous-trames du G.729 ne coïncident pas avec celles du G.723.1, mais qu'au contraire les secondes (7,5 ms) chevauchent les premières (5 ms). La figure 5b représente une trame du codage G723.1 et trois trames de codage G.729 et leurs sous-trames respectives. Les sous-trames de la trame du G.723.1 sont numérotées de 0 à 3. Les trois trames du G.729 sont regroupées et leurs sous-trames sont numérotées de 0 à 5.

### Détermination du retard du filtre multitap

[0075] La détermination du retard est directe. Ainsi pour les sous-trames paires du G723.1, c'est-à-dire les sous-trames 0 et 2, le retard est pris égal à la partie entière de celui des sous-trames 1 et 4 du G.729. Pour les sous-trames impaires, une boucle fermée est effectuée autour du retard précédent (sous-trame paire). Cette boucle fermée peut être identique à celle du G.723.1, mais peut aussi être restreinte selon la complexité désirée, voire éliminée pour garder alors la même valeur de retard sur les deux sous-trames paires et impaires.

### Détermination des coefficients du filtre multitap

[0076] Ici, on ne considère qu'un seul premier dictionnaire qui est l'ensemble des 128 gains LTP monotap du G.729 tandis que l'on considère deux seconds dictionnaires possibles (les deux dictionnaires de vecteurs de gains du G.723.1 dont le choix dépend du retard des sous-trames).

[0077] Une fois le retard déterminé, il reste encore à déterminer un vecteur de 5 gains dans le dictionnaire de vecteurs de 5 coefficients que sélectionne le codeur G.723.1. La mise en oeuvre de la présente invention permet d'en restreindre l'exploration à un nombre limité de vecteurs de gains déterminés à partir des gains LTP monotap des sous-trames du codeur G.729.

5 [0078] Au préalable, une étude statistique a été réalisée en associant au sein d'un même codeur le modèle multitap du codeur G.723.1 et le modèle monotap du codeur G.729. Cette étude a permis pour chacun des 128 gains LTP monotap du G.729 de classer les 170 et 85 vecteurs de gains LTP multitap des deux dictionnaires du G.723.1 selon leur impact sur la qualité du signal restitué. Ici, c'est le critère CELP qui est utilisé à cette fin. Pour chacun de ces deux dictionnaires du G.723.1, on a ainsi obtenu 128 ordres (ou classements) associés à la partition élémentaire de l'ensemble des 128 gains LTP monotap.

10 [0079] Chaque sous-trame du G.723.1 couvre (au moins partiellement) deux sous-trames du G.729. Tout d'abord, on extrait les deux gains monotap (notés  $g_1$  et  $g_2$ ) de ces deux sous-trames correspondantes du G.729. A chacun de ces deux gains, est associé un classement  $C(g_j)$  des vecteurs du dictionnaire de vecteurs de coefficients multitap. Ce dictionnaire est sélectionné par la valeur du retard de la sous-trame paire du G.723.1.

15 [0080] Soit  $N$  le nombre maximal, autorisé, de vecteurs de gains multitap pour la sous-trame courante du codeur G.723.1. Si les deux gains du G.729 sont égaux, il n'y a donc qu'un seul classement et les  $N$  premiers éléments ordonnés par ce classement du dictionnaire de vecteurs de gains sont retenus. Sinon, on constitue un ordre de  $N$  éléments à partir de deux ordres différents. Par exemple, on constitue deux sous-ensembles des classements  $C(g_1)$  et  $C(g_2)$  en présélectionnant leurs  $N_1$  et  $N_2$  (respectivement) premiers éléments.  $N_1$  et  $N_2$  sont inférieurs ou égaux à  $N$ . On peut traiter équitablement les deux classements ( $N_1=N_2$ ) ou privilégier l'un des deux classements. Par exemple, on peut privilégier le classement associé au gain monotap le plus grand (typiquement si  $g_1 > g_2$  alors  $0 \leq N_2 < N_1 \leq N$ ). On peut aussi privilégier celui dont la sous-trame du G.729 recouvre le plus la sous-trame du G.723.1 considérée. Puis, on sélectionne tout d'abord tous les éléments appartenant aux deux sous-ensembles. On complète l'ensemble formant le dictionnaire, à  $N$ , en prenant alternativement dans les deux sous-ensembles l'élément le mieux classé parmi les restants. Là encore, on peut, en complétant, favoriser l'un des deux sous-ensembles. On peut, bien entendu, combiner certaines de ces stratégies. Par exemple, choisir  $N_1=N_2$  mais après sélection des éléments communs, poursuivre par les éléments restants de l'un des deux classements avant de compléter éventuellement par les éléments restants de l'autre classement. La stratégie peut aussi varier selon la sous-trame du G.723.1 considérée.

25 [0081] Finalement, l'exploration du dictionnaire de vecteurs de gains est limitée aux  $N$  vecteurs déterminés grâce à l'ordre "dynamique" ainsi constitué. Cette exploration focalisée permet de sélectionner le meilleur vecteur de gains. Préférentiellement, le critère de sélection est le critère CELP utilisé classiquement par le G.723.1 pour l'exploration des dictionnaires des vecteurs à 5 coefficients LTP. La solution exposée ici permet une très forte réduction de la complexité de l'analyse LTP du codage G.723.1 sans pour autant nuire à la qualité. A titre d'exemple de performances, on a représenté sur les figures 9a et 9b, pour les deux dictionnaires, l'histogramme des tailles d'exploration qui garantissent une perte sur le critère CELP strictement inférieure à 1% par rapport à une exploration complète. On note que les tailles d'exploration (en abscisse) sont bien inférieures à la taille totale du dictionnaire. Ainsi, la taille moyenne est de 39 pour le dictionnaire à 85 vecteurs et de 49 pour le dictionnaire à 170 vecteurs. Sur la base d'apprentissage utilisée, l'étude statistique montre, même pour des tailles d'exploration moyennes, bien inférieures aux tailles des dictionnaires (48 au lieu de 85 et 58 au lieu de 170), que l'exploration restreinte est optimale selon le critère CELP (pratiquement pas de perte sur le critère CELP). La recherche focalisée peut donc mener à des performances équivalentes à la recherche exhaustive tout en n'explorant qu'à peine plus de la moitié du dictionnaire de taille 85 et le tiers du dictionnaire de taille 170. Ces chiffres illustrent bien la réduction de complexité qu'apporte la mise en oeuvre de la présente invention.

30 [0082] Par ailleurs, un stockage complet des 128 ordres pour les deux dictionnaires représente un total de  $128 \times (170+85) = 32640$  valeurs d'index à stocker. En réalité, il n'est pas nécessaire de retenir toutes ces valeurs puisque, comme indiqué ci-dessus, seul un nombre limité est nécessaire. Ainsi, pour une perte nulle sur le critère CELP, les essais montrent qu'il suffirait de stocker environ 13582 indices. En choisissant une contrainte moins forte sur le critère CELP, on peut encore réduire ce nombre (jusqu'à atteindre 11251 valeurs pour 1% de perte). Il peut encore être fortement réduit en adoptant pour l'ensemble des gains monotap une autre partition que la partition élémentaire.

50 ***Deuxième exemple de réalisation : détermination des paramètres LTP monotap du G.729 à 8 kbit/s à partir des paramètres LTP multitap du G.723.1 à 6,3 kbit/s***

[0083] A l'inverse du mode de réalisation précédent, on dispose des paramètres du modèle LTP multitap d'une trame de G.723.1 et on cherche à obtenir les paramètres LTP monotap du G.729 pour trois trames c'est-à-dire six sous-trames (voir figure 5b).

**Détermination du retard en boucle ouverte**

**[0084]** La recherche en boucle ouverte a été éliminée. Pour cela, chacune des trois trames G.729 adopte tout d'abord pour retard en boucle ouverte le retard de l'une des sous-trames du codeur G.723.1. La correspondance entre trames G.729 et sous-trame G.723.1 est illustrée sur la figure 6.

**[0085]** Toutefois, il est à noter que le retard choisi par le codeur G.723.1 peut être hors de la plage de valeurs autorisées par le codeur G.729. En effet, la plus petite valeur autorisée par le codeur G.729 est 19 alors qu'elle est de 18 pour le codeur G.723.1. Plusieurs solutions sont possibles pour contourner ce problème. Typiquement, on peut par exemple doubler le retard issu du codeur G.723.1, ou plus simplement lui ajouter 1.

**Détermination du retard en boucle fermée**

**[0086]** Une fois les retards en boucle ouverte fixés pour les trois trames du codeur G.729, il reste à effectuer pour chaque sous-trame la recherche en boucle fermée. On rappelle que les plages de valeurs sont les suivantes :

$$\lambda_0 \in [\lambda_{OL}-3; \lambda_{OL}+3] \quad \text{et} \quad \lambda_1 \in [\text{int}(\lambda_0)-5\frac{2}{3}; \text{int}(\lambda_0)+4\frac{2}{3}]$$

**[0087]** La recherche en boucle fermée standard du codeur G.729 consiste tout d'abord à tester successivement toutes les valeurs entières de la plage (7 valeurs pour  $\lambda_0$  et 10 pour  $\lambda_1$ ). Une fois sélectionnée la meilleure valeur entière, les différentes fractions (-2/3, -1/3, 1/3, 2/3) sont testées pour déterminer la meilleure selon le critère choisi, en l'espèce celle qui maximise le critère CELP. Pour la sous-trame paire, on notera que la partie fractionnaire n'est recherchée que si la partie entière de  $\lambda_0$  est inférieure à 85.

**[0088]** Ici, le premier dictionnaire (dans la définition de l'invention donnée ci-avant) est l'un des deux dictionnaires de vecteurs de gains LTP du codeur G.723.1, le second dictionnaire étant l'un des deux ensembles des valeurs entières de voisinage (ou gigue) autour d'un retard d'ancrage. On comprendra alors que l'invention peut s'appliquer aisément à plus d'un premier dictionnaire, d'une part, et à plus d'un second dictionnaire, d'autre part.

**[0089]** Pour réduire la complexité de la recherche en boucle fermée des valeurs entières au voisinage de la valeur d'ancrage  $\lambda'$  ( $\lambda_{OL}$  ou  $\text{int}(\lambda_0)$ ), on propose, au sens de l'invention, de limiter le nombre de valeurs entières de retard testées par les boucles fermées. En fonction du choix du vecteur de gains LTP fait par le G.723.1, seul un nombre réduit de valeurs est testé. Le retard entier est déterminé dans cet ensemble restreint. Puis, la partie fractionnaire est recherchée de façon classique.

**[0090]** Auparavant, une étude statistique a été réalisée en associant au sein d'un même codeur le modèle multitap du G.723.1 et le modèle monotap du G.729. Cette étude a permis, pour chacun des vecteurs de gains des deux dictionnaires LTP multitap du G.723.1, d'établir pour les deux voisinages de recherche en boucle fermée du G.729 (sous-trames paire et impaire) un ordre d'importance des valeurs de voisinage selon leur impact sur la qualité du signal restitué. Cette classification permet de choisir le nombre de valeurs testées selon les contraintes de qualité et de complexité et de limiter, pour chacune des six sous-trames du G.729, l'étendue de la boucle fermée à partir du choix des gains ( $\beta_j$ ) fait pour les sous-trames du G.723.1. En utilisant la correspondance entre sous-trames du tableau de la figure 8, chaque sous-trame G.729 est associée à une ou deux sous-trames G.723.1. A partir du vecteur de 5 coefficients du vecteur de gains ( $\beta_j$ ), les valeurs de voisinage de  $\lambda'$  sont classées par ordre d'importance décroissante. Le nombre de valeurs testées est alors déterminé en fonction de la complexité visée ou du rapport qualité/complexité visé.

**[0091]** L'association entre sous-trames paires (respectivement impaires) du codeur G.729 et le jeu de paramètres ( $\lambda_j$ ,  $(\beta_j)_j$ ) issu du codeur G.723.1 est illustrée sur la figure 7a (respectivement sur la figure 7b).

**[0092]** On notera que pour certaines sous-trames, la valeur d'ancrage  $\lambda'$  peut être différente du retard  $\lambda_j$  du jeu de paramètres ( $\lambda_j, (\beta_j)_j$ ) déterminé pour la sous-trame G.723.1 associée. Ce point est explicité plus loin où l'on prend en compte la parité des sous-trames (paires ou impaires). Dans une première variante, on peut simplement ne pas tenir compte d'une différence éventuelle. Avantagusement, dans une autre variante, l'ensemble des voisinages ordonnés est modifié en fonction de la différence ( $\lambda_j - \lambda'$ ) et la taille de cet ensemble est éventuellement modifiée. Préférentiellement, on soustrait la différence ( $\lambda_j - \lambda'$ ) à chaque élément de ce voisinage ordonné selon les gains ( $\beta_j$ ) et on considère son intersection avec l'ensemble de définition des voisinages (ici l'intervalle [-3;3] pour les sous-trames paires et l'intervalle [-5,4] pour les sous-trames impaires, comme on le verra plus loin).

**[0093]** On peut aussi conditionner l'utilisation des voisinages restreints en fonction de l'écart entre les deux retards. La stratégie peut donc être adaptée à la sous-trame ou à l'écart entre les retards, ou aux deux critères combinés.

Sous-trames paires

**[0094]** La recherche doit être effectuée autour du retard en boucle ouverte  $\lambda_{OL}$  sur la plage  $[\lambda_{OL}-3; \lambda_{OL}+3]$ . En fonction du (des) vecteurs de gains choisi(s) par le codeur G.723.1, des ordres de l'ensemble des 7 valeurs de gigue (-3, -2, -1, 0, 1, 2, 3) sont déterminés. Pour la sous-trame 0 (respectivement 2) du codeur G.729, il n'y a qu'une seule sous-trame du G.723.1 associée et donc un seul vecteur de gains et, ainsi, un seul ordre. Par contre, deux sous-trames du codeur G.723.1 sont associées à la sous-trame 4 du codeur G.729, comme le montre la figure 7a. Deux ordres de l'ensemble des voisinages sont donc présélectionnés par les vecteurs de gains  $(\beta_i)_2$  et  $(\beta_i)_3$ . Comme indiqué ci-avant, on peut retenir un seul ordre ou combiner les 2 ordres. Si on ne retient que l'ordre associé au vecteur  $(\beta_i)_3$ , ou si l'on fixe  $\lambda_2=\lambda_3$  (où  $\lambda_3$  est la valeur d'ancrage), aucun traitement particulier n'est effectué. Autrement, l'ensemble ordonné des 7 voisinages correspondant à  $(\beta_i)_2$  est modifié en fonction de  $(\lambda_2-\lambda_3)$ . Puis, on complète éventuellement avec l'ensemble ordonné selon  $(\beta_i)_3$ . Les N premiers éléments selon l'ordre obtenu sont testés, la taille N ( $N \leq 7$ ) est définie en fonction de la complexité ou du compromis qualité/complexité visé.

Sous-trames impaires

**[0095]** La recherche doit être menée autour de la partie entière  $\lambda'_{2p}$  de la sous-trame (paire) précédente sur la plage  $[\lambda'_{2p}-5_{2/3}; \lambda'_{2p}+4_{2/3}]$ . Pour ces sous-trames impaires, comme pour la sous-trame paire 4, le retard  $\lambda_j$  du jeu de paramètres  $(\lambda_j, (\beta_i)_j)$  de (ou des) sous-trames G.723.1 associé(es) peut être différent de cette valeur d'ancrage  $\lambda'_{2p}$ . En fonction du (des) vecteur(s)  $(\beta_i)_j$  de gains choisi(s) par le codeur G.723.1, des ordres de l'ensemble des 10 valeurs de gigue sont présélectionnés et modifiés en fonction de la différence  $(\lambda_j-\lambda'_{2p})$ . Soit N ( $N \leq 10$ ) le nombre maximal autorisé de valeurs testées.

**[0096]** Pour déterminer la plage de recherche restreinte, on procède préférentiellement comme suit pour chaque sous-trame impaire.

*Sous-trame 1 :*

**[0097]** La plage totale de recherche est  $[\lambda'_0-5_{2/3}; \lambda'_0+4_{2/3}]$ . Deux ordres correspondants aux vecteurs de gains  $(\beta_i)_0$  et  $(\beta_i)_1$  sont présélectionnés. Puis, les voisinages ordonnés sont modifiés en fonction des différences  $(\lambda_0-\lambda'_0)$  et  $(\lambda_1-\lambda'_0)$ . Ces deux écarts sont limités car :

- d'une part, le retard  $\lambda'_0$  en boucle fermée du G.729 est dans le voisinage (dans l'intervalle [-3,3]) du retard en boucle ouverte (ici, pris égal à  $\lambda_0$  correspondant au retard en boucle fermée du G.723.1),
- d'autre part, dans le codeur G.723.1, l'écart entre les retards en boucle fermée d'une sous-trame paire et la sous-trame impaire suivante est limité car la différence  $(\lambda_1-\lambda_0)$  est dans l'intervalle [-1, 2].

**[0098]** A partir des  $N_1$  et  $N_2$  premiers éléments des voisinages modifiés, on constitue un voisinage ordonné unique de taille N. On sélectionne tout d'abord les valeurs communes aux deux sous-ensembles, puis on complète l'ensemble, si nécessaire, en prenant alternativement la meilleure valeur restante dans les deux sous-ensembles. La recherche en boucle fermée est alors menée dans le sous-ensemble ainsi constitué.

*Sous-trame 3 :*

**[0099]** La plage totale de recherche est  $[\lambda'_2-5_{2/3}; \lambda'_2+4_{2/3}]$ . Un ordre correspondant au vecteur de gains  $(\beta_i)_2$  est sélectionné. Puis, le voisinage ordonné est modifié en fonction de la différence  $(\lambda_2-\lambda'_2)$ . Contrairement au cas précédent, l'écart entre  $\lambda_2$  et  $\lambda'_2$  peut être important et l'intersection du voisinage ordonné, modifié en soustrayant  $(A_2-A'_2)$ , peut être nulle. Dans ce cas, préférentiellement, la recherche est faite sur toute la plage  $[\lambda'_1-5_{2/3}; \lambda'_1+4_{2/3}]$ . L'utilisation de voisinages ordonnés peut aussi être conditionnée à un seuil sur  $|\lambda_2-\lambda'_2|$ . Par exemple, les voisinages ne sont restreints que si  $|\lambda_2-\lambda'_2| < 3$  ; autrement toute la plage [-5,4] est explorée. Le choix de cette variante peut aussi dépendre de la complexité autorisée.

*Sous-trame 5 :*

**[0100]** La plage totale de recherche est  $[\lambda'_4-5_{2/3}; \lambda'_4+4_{2/3}]$ . Un ordre correspondant au vecteur de gains  $(\beta_i)_3$  est sélectionné. Puis, le voisinage ordonné est modifié en fonction de la différence  $(\lambda_3-\lambda'_4)$ . Comme dans le cas de la sous-trame 1, cet écart est limité. En effet, le retard en boucle fermée du G.729,  $\lambda'_2$ , est dans le voisinage ([-3,3]) du retard en boucle ouverte (ici pris égal au retard en boucle fermée  $\lambda_3$  du G.723.1). On explore les N premières valeurs de

l'ensemble ordonné modifié.

[0101] La solution présentée ici permet une très forte réduction de la complexité de l'analyse LTP du codage G.729. Par rapport à l'exploration des voisinages complets, l'invention permet de ne tester que 60% (respectivement 40 %) des valeurs de voisinage si le vecteur de gains du codeur G.723.1 est dans le dictionnaire à 170 entrées (respectivement 85 entrées).

***Troisième exemple de réalisation : détermination des paramètres LTP multitaq du codeur G.723.1 à 5,3 kbit/s à partir des paramètres LTP multitaq du codeur G.723.1 à 6,3 kbit/s***

[0102] Les deux modèles sont très proches et ne se distinguent pratiquement que par le choix du dictionnaire des vecteurs de gains LTP multitaq.

**Détermination du retard du filtre multitaq**

[0103] De façon similaire à la détermination du retard d'un monotap décrit ci-avant à partir des paramètres LTP multitaq, on peut utiliser le retard des sous-trames paires, en tant que retard en boucle ouverte de la super sous-trame, puis restreindre la plage de variation du retard en boucle fermée du mode à 5,3 kbit/s en fonction du vecteur de cinq coefficients du filtre choisi par le mode à 6,3 kbit/s. Préférentiellement, aucun traitement autre qu'une simple recopie du retard n'est nécessaire. Ainsi, chaque sous-trame du 5,3 kbit/s adopte pour retard celui que le mode 6,3 kbit/s a choisi pour la même sous-trame.

**Détermination des coefficients du filtre multitaq**

[0104] Ici, on a un seul second dictionnaire qui est le dictionnaire à 170 vecteurs de cinq coefficients du mode à 5,3 kbit/s tandis qu'il faut considérer deux "*premiers dictionnaires*", selon la terminologie utilisée dans la définition générale de l'invention. Ces deux premiers dictionnaires sont les deux dictionnaires de vecteurs de gains utilisés par le mode à 6,3 kbit/s du G.723.1.

[0105] Dans cet exemple de réalisation, on cherche donc à déterminer pour le mode à 5,3 kbit/s un vecteur de gains dans le dictionnaire à 170 entrées à partir d'un vecteur de gains sélectionné par le mode 6,3 kbit/s dans l'un des deux dictionnaires (à 170 ou à 85 vecteurs).

[0106] L'un des deux cas peut sembler trivial car si le mode 6,3 kbit/s utilise pour la sous-trame courante le même dictionnaire (le dictionnaire à 170 vecteurs), on peut être tenté de choisir pour le mode 5,3 kbit/s le même vecteur que le mode à 6,3 kbit/s. Néanmoins, cette approche apporte une dégradation sensible du signal. En effet, bien que la modélisation LTP soit identique pour les deux modes (même dictionnaires de retards et de vecteurs de 5 gains), il faut garder à l'esprit que le reste du processus de codage n'est pas le même. Le filtrage LTP n'est donc pas appliqué au même signal et il faut alors élargir le choix de vecteurs de coefficients du filtre pour le mode 5,3 kbit/s.

[0107] A cet effet, une étude a été réalisée sur les deux dictionnaires pour associer à chacun des vecteurs, un classement des vecteurs du dictionnaire à 170 vecteurs.

[0108] Ainsi, pour sélectionner un vecteur de gains pour le mode 5,3 kbit/s, il est préféré, à partir du choix du vecteur de gains fait par le mode 6,3 kbit/s, de n'explorer dans le grand dictionnaire (170 vecteurs) qu'un ensemble restreint aux N premiers vecteurs du classement associé au vecteur de gain choisi par le mode 6,3 kbit/s. La taille N dépend de la complexité ou de la qualité ou du compromis/qualité complexité désiré. Dans ce sous-ensemble, on sélectionne alors le vecteur de gains qui maximise un critère, préférentiellement le critère CELP, comme décrit ci-avant.

**Revendications**

1. Procédé de codage d'un signal audio selon un second format, à partir d'informations obtenues par la mise en oeuvre d'au moins une étape de codage selon un premier format, les premier et second formats mettant en oeuvre, notamment pour le codage d'un signal de parole, une étape de recherche de paramètres de prédiction à long terme LTP par exploration d'au moins un dictionnaire comportant des paramètres candidats, l'un au moins des premier et second format de codage utilisant un filtrage à plusieurs coefficients pour une recherche fine des paramètres LTP, **caractérisé en ce qu'il** comporte les étapes suivantes :

- accéder à des résultats d'une étude statistique et/ou analytique, menée en fonction de jeux successifs de paramètres LTP selon le premier format de codage, pour déterminer un nombre d'ordres et des ordres appropriés dans un dictionnaire qu'utilise le second format de codage,
- récupérer une information a priori de partition du premier dictionnaire concernant une classe de la partition à



## EP 1 836 699 B1

laquelle appartient un paramètre LTP obtenu au cours du codage selon le premier format, obtenue suite à la détermination des paramètres LTP au cours du codage selon le premier format, pour sélectionner au moins un ordre dudit dictionnaire qu'utilise le second format de codage,

- appliquer l'ordre sélectionné aux candidats dudit dictionnaire qu'utilise le second format de codage pour choisir un nombre limité de premiers candidats, et  
- pour effectuer le second codage, mener la recherche LTP uniquement parmi ledit nombre limité de candidats.

2. Procédé selon la revendication 1, **caractérisé en ce que** l'on prévoit initialement une partition élémentaire du premier dictionnaire comportant N éléments, en N classes disjointes de taille 1.

3. Procédé selon la revendication 1, dans lequel le premier format de codage utilise un premier dictionnaire et le second format de codage utilise un second dictionnaire, **caractérisé en ce que** l'on prévoit une partition du premier dictionnaire en classes non disjointes, de sorte qu'un même élément peut être associé à plus d'un ordre du second dictionnaire.

4. Procédé selon l'une des revendications 1 à 3, dans lequel le premier format de codage utilise un premier dictionnaire et le second format de codage utilise un second dictionnaire, **caractérisé en ce que** l'on prévoit un regroupement d'ordres similaires pour modifier dynamiquement la partition initiale du premier dictionnaire et, de là, le nombre d'ordres du second dictionnaire.

5. Procédé selon la revendication 4, **caractérisé en ce que** l'on prévoit en outre une opération consistant à recalculer successivement les ordres du second dictionnaire, une fois regroupés, et **en ce que** l'on modifie dynamiquement la partition initiale du premier dictionnaire et/ou les ordres ainsi regroupés.

6. Procédé selon l'une des revendications 4 à 5, dans lequel le premier format de codage utilise un premier dictionnaire et le second format de codage utilise un second dictionnaire, **caractérisé en ce que**, pour chacun des ordres du second dictionnaire, un nombre maximal d'éléments du second dictionnaire à retenir est choisi en fonction des classes du premier dictionnaire et/ou des ordres du second dictionnaire, pour limiter une ressource en mémoire utilisée pour le stockage des ordres du second dictionnaire.

7. Procédé selon l'une des revendications précédentes, **caractérisé en ce que** ledit nombre limité de candidats est choisi en fonction d'un compromis entre qualité et complexité du second codage.

8. Procédé selon la revendication 7 dans lequel un signal d'entrée à coder est traité par blocs de données, **caractérisé en ce que** ledit compromis est fixé dynamiquement, à chaque bloc de données à traiter en fonction de paramètres du premier format de codage et/ou de caractéristiques du signal à coder et préférentiellement en fonction de sous-trames LTP que comporte chaque bloc de données.

9. Procédé selon l'une des revendications 1 à 8, dans lequel un signal d'entrée à coder est traité par blocs de données comportant chacun, pour le premier format de codage, des premières sous-trames LTP et, pour le second format de codage, des secondes sous-trames LTP, **caractérisé en ce que**, pour des premières et secondes sous-trames de durée identique, à chaque sous-trame courante du deuxième format de codage correspond une seule sous-trame du premier format de codage, et **en ce que** :

- le premier format de codage sélectionne un premier jeu de paramètres LTP pour la sous-trame courante,  
- à partir de la partition par classes du dictionnaire associé à l'un des paramètres LTP du premier format, on sélectionne un ordre d'exploration du dictionnaire du second format en choisissant un ordre associé à la classe de l'élément dudit premier jeu, et  
- suivant l'ordre ainsi sélectionné, un nombre limité de premiers candidats du dictionnaire du second format est exploré.

10. Procédé selon l'une des revendications 1 à 8, dans lequel un signal d'entrée à coder est traité par blocs de données comportant chacun, pour le premier format de codage, des premières sous-trames LTP et, pour le second format de codage, des secondes sous-trames LTP, **caractérisé en ce que**, pour des premières et secondes sous-trames de durées différentes :

- le premier format de codage sélectionne une pluralité de jeux de paramètres LTP, pour des premières sous-trames correspondant sensiblement une seconde sous-trame courante,

- à partir de la partition par classes du dictionnaire associé à l'un des paramètres LTP du premier format, on présélectionne des ordres d'exploration du dictionnaire du second format en choisissant les ordres associés aux classes des éléments desdits jeux de paramètres LTP,
- à partir de la présélection desdits ordres, on détermine au moins un ordre préféré, et
- ledit dictionnaire du second format est exploré suivant l'ordre préféré, en se limitant à ses premiers éléments.

11. Procédé selon l'une des revendications précédentes, dans lequel le premier format de codage utilise un filtrage à coefficient unique pour des premières sous-trames LTP tandis que le second format de codage utilise un filtrage à plusieurs coefficients pour des secondes sous-trames LTP, **caractérisé en ce que** :

- pour chaque première sous-trame, par la mise en oeuvre du premier format de codage, on détermine un couple  $(\lambda_e, \beta_e)$  de premiers paramètres du filtre LTP à coefficient unique,
- pour le codage d'une seconde sous-trame courante, on détermine une pluralité de couples  $(\lambda_s, (\beta_i)_s)$  de paramètres du filtre LTP à plusieurs coefficients sur la base du jeu de paramètres du premier format  $(\lambda_e, \beta_e)$ , avec :

- une détermination d'un retard LTP  $\lambda_s$  correspondant préférentiellement à celui déterminé par le premier format de codage sur une première sous-trame qui recouvre le plus la seconde sous-trame courante,
- une détermination d'un vecteur de gains  $(\beta_i)_s$  pour la seconde sous-trame courante à partir de l'un au moins des gains  $\beta_e$  des premières sous-trames, par la mise en oeuvre des étapes b), c) et d) où les ordres du dictionnaire du second format correspondent à un ensemble de vecteurs de gains  $(\beta_i)_s$  de la seconde sous-trame.

12. Procédé selon la revendication 11, **caractérisé en ce que**, pour le codage d'une seconde sous-trame courante :

- à partir de premiers gains LTP du premier format  $(\beta_e)$  choisis pour une ou plusieurs premières sous-trames correspondant à une seconde sous-trame courante, les ordres du dictionnaire du deuxième format, associés à des classes des premiers gains LTP, sont présélectionnés,
- un seul de ces ordres est constitué, de préférence dynamiquement, à partir desdits ordres présélectionnés pour ladite seconde sous-trame courante, et
- N premiers vecteurs de seconds gains, déterminés par l'ordre constitué, sont testés pour sélectionner, selon un critère choisi, un meilleur vecteur de gains à associer à la seconde sous-trame.

13. Procédé selon l'une des revendications 1 à 10, dans lequel le second format de codage utilise un filtrage à coefficient unique pour des secondes sous-trames LTP tandis que le premier format de codage utilise un filtrage à plusieurs coefficients pour des premières sous-trames LTP, **caractérisé en ce que** :

- pour chaque première sous-trame, par la mise en oeuvre du premier format de codage, on détermine un premier jeu de paramètres LTP  $\lambda_e, (\beta_i)_e$  correspondant à un couple comportant un retard LTP  $\lambda_e$  et un vecteur de gains associés  $(\beta_i)_e$  du filtre LTP à plusieurs coefficients,
- on effectue une partition d'un dictionnaire des vecteurs de gains  $(\beta_i)_e$  du premier format,
- pour le codage d'une seconde sous-trame courante par le second format, on détermine des ordres d'un dictionnaire du second format pour des premières sous-trames correspondant à la seconde sous-trame courante, ledit dictionnaire du second format étant constitué d'un ensemble de valeurs de gigue et lesdits ordres de ce dictionnaire étant associés à la partition du dictionnaire du premier format,
- on détermine un ordre des valeurs de gigue et on explore successivement des valeurs de retards LTP pour le second format, sur les valeurs de gigue ainsi ordonnées et autour d'un ou plusieurs retards d'ancrage déterminés en fonction des retards  $\lambda_e$  sur les premières sous-trames.

14. Procédé selon l'une des revendications 1 à 10, dans lequel le premier format de codage utilise un filtrage à plusieurs coefficients sur des premières sous-trames LTP et le second format de codage utilise un filtrage à plusieurs coefficients sur des secondes sous-trames LTP, **caractérisé en ce que** :

- à partir d'au moins un premier jeu de paramètres sélectionné par le premier format et incluant au moins un vecteur de gains  $(\beta_i)_e$  déterminé pour au moins une première sous-trame, on mène une partition du dictionnaire du premier format correspondant à un dictionnaire des vecteurs de gains du premier format  $(\beta_i)_e$ ,
- on en déduit des ordres du dictionnaire du second format correspondant à un dictionnaire des vecteurs de gains  $(\beta_i)_s$  du second format, lesdits ordres étant associés à ladite partition,
- à partir de vecteurs de gains  $(\beta_i)_e$  choisis par le premier format pour des premières sous-trames qui couvrent

sensiblement la seconde sous-trame courante, des ordres du deuxième dictionnaire associés à des classes de ladite partition sont présélectionnés,

- on retient l'un des ordres présélectionnés,

- on détermine plusieurs vecteurs de gains à associer à la seconde sous-trame courante, en fonction de l'ordre retenu, et

- on sélectionne, par tests sur lesdits plusieurs vecteurs de gain, le meilleur vecteur de gain selon un critère choisi.

15. Dispositif de codage d'un signal audio selon un second format, agencé pour utiliser des informations de codage obtenues par la mise en oeuvre d'un codage selon un premier format, les premier et second formats mettant en oeuvre, notamment pour le codage d'un signal de parole, une recherche de paramètres de prédiction à long terme LTP par exploration d'un dictionnaire comportant des paramètres candidats, l'un au moins des premier et second format de codage utilisant un filtrage à plusieurs coefficients pour une recherche fine des paramètres LTP, **caractérisé en ce qu'il** comporte :

- une mémoire stockant une table de correspondance définissant, en fonction de paramètres LTP déterminés par le premier format de codage, des ordres d'un dictionnaire qu'utilise le second format de codage, la table de correspondance étant définie à partir de résultats d'une étude statistique et/ou analytique, menée en fonction de jeux successifs de paramètres LTP selon le premier format de codage, pour déterminer un nombre d'ordres et des ordres appropriés dans un dictionnaire qu'utilise le second format de codage,

- des moyens pour récupérer un signal donnant au moins une information a priori de partition du premier dictionnaire concernant une classe de la partition à laquelle appartient un paramètre LTP obtenu au cours du codage selon le premier format, obtenue suite à la détermination des paramètres LTP au cours d'un codage selon le premier format, pour sélectionner au moins un ordre dudit dictionnaire qu'utilise le second format de codage,

- des moyens actifs sur réception dudit signal pour consulter ladite table de correspondance et sélectionner au moins un ordre dudit dictionnaire qu'utilise le second format de codage,

- des moyens de calcul pour :

- ordonner ledit dictionnaire qu'utilise le second format de codage selon l'ordre sélectionné, en vue de choisir un nombre limité de premiers candidats dans le dictionnaire, et

- poursuivre le codage selon le second format, en menant la recherche LTP uniquement parmi ce nombre limité de candidats

16. Système de codage mettant en oeuvre au moins un premier et un second format de codage, **caractérisé en ce qu'il** comporte au moins un dispositif de codage selon le premier format et un dispositif de codage selon la revendication 15, appliquant ledit second format.

17. Produit programme d'ordinateur, stocké dans une mémoire d'une unité de traitement ou sur un support amovible destiné à coopérer avec un lecteur de ladite unité de traitement ou téléchargeable d'un site distant, **caractérisé en ce qu'il** comporte des instructions pour la mise en oeuvre de tout ou partie des étapes du procédé selon l'une des revendications 1 à 14.

## Claims

1. Method for coding an audio signal according to a second format, on the basis of information obtained by implementing at least one step of coding according to a first format, the first and second formats implementing, in particular for the coding of a speech signal, a step of searching for LTP long-term prediction parameters by exploring at least one dictionary comprising candidate parameters, one at least of the first and second coding formats using a filtering with several coefficients for a fine search for the LTP parameters, **characterized in that** it comprises the following steps:

- accessing results from a statistical and/or analytical study, conducted as a function of successive suites of LTP parameters according to the first coding format, so as to determine a number of orders and appropriate orders in a dictionary that the second coding format uses,

- recovering an *a priori* information of partition of the first dictionary relating to a class of the partition to which an LTP parameter obtained in the course of the coding according to the first format belongs, obtained following the determination of the LTP parameters in the course of the coding according to the first format, so as to select at least one order of said dictionary that the second coding format uses,

- applying the selected order to the candidates of said dictionary that the second coding format uses so as to choose a limited number of first candidates, and
- so as to perform the second coding, conducting the LTP search only among said limited number of candidates.

- 5     **2.** Method according to Claim 1, **characterized in that** an elementary partition of the first dictionary comprising N elements, into N disjoint classes of size 1, is initially envisaged.
- 10     **3.** Method according to Claim 1, in which the first coding format uses a first dictionary and the second coding format uses a second dictionary, **characterized in that** a partition of the first dictionary into non-disjoint classes is envisaged, so that one and the same element can be associated with more than one order of the second dictionary.
- 15     **4.** Method according to one of Claims 1 to 3, in which the first coding format uses a first dictionary and the second coding format uses a second dictionary, **characterized in that** a grouping of similar orders is envisaged so as to dynamically modify the initial partition of the first dictionary and, thereby, the number of orders of the second dictionary.
- 20     **5.** Method according to Claim 4, **characterized in that** furthermore an operation is envisaged consisting in successively recalculating the orders of the second dictionary, once grouped together, and **in that** the initial partition of the first dictionary and/or the orders thus grouped together is dynamically modified.
- 25     **6.** Method according to either of Claims 4 and 5, in which the first coding format uses a first dictionary and the second coding format uses a second dictionary, **characterized in that**, for each of the orders of the second dictionary, a maximum number of elements of the second dictionary to be retained is chosen as a function of the classes of the first dictionary and/or of the orders of the second dictionary, so as to limit a memory resource used for storing the orders of the second dictionary.
- 30     **7.** Method according to one of the preceding claims, **characterized in that** said limited number of candidates is chosen as a function of a compromise between quality and complexity of the second coding.
- 35     **8.** Method according to Claim 7, in which an input signal to be coded is processed by data blocks, **characterized in that** said compromise is fixed dynamically, with each data block to be processed as a function of parameters of the first coding format and/or of characteristics of the signal to be coded and preferably as a function of LTP subframes that each data block comprises.
- 40     **9.** Method according to one of Claims 1 to 8, in which an input signal to be coded is processed by data blocks each comprising, for the first coding format, first LTP subframes and, for the second coding format, second LTP subframes, **characterized in that**, for first and second subframes of identical duration, to each current subframe of the second coding format there corresponds a single subframe of the first coding format, and **in that**:
- the first coding format selects a first suite of LTP parameters for the current subframe,
  - on the basis of the partition by classes of the dictionary associated with one of the LTP parameters of the first format, an order of exploration of the dictionary of the second format is selected by choosing an order associated with the class of the element of said first suite, and
  - following the order thus selected, a limited number of first candidates of the dictionary of the second format is explored.
- 45     **10.** Method according to one of Claims 1 to 8, in which an input signal to be coded is processed by data blocks each comprising, for the first coding format, first LTP subframes and, for the second coding format, second LTP subframes, **characterized in that**, for first and second subframes of different durations:
- the first coding format selects a plurality of LTP parameter suites, for first subframes corresponding substantially to a second current subframe,
  - on the basis of the partition by classes of the dictionary associated with one of the LTP parameters of the first format, orders of exploration of the dictionary of the second format are preselected by choosing the orders associated with the classes of the elements of said LTP parameter suites,
  - at least one preferred order is determined on the basis of the preselection of said orders, and
  - said dictionary of the second format is explored following the preferred order, limiting the exploration to its first elements.
- 50
- 55

11. Method according to one of the preceding claims, in which the first coding format uses a filtering with one coefficient for first LTP subframes while the second coding format uses a filtering with several coefficients for second LTP subframes, **characterized in that:**

- 5 - for each first subframe, by implementing the first coding format, a pair  $(\lambda_e, \beta_e)$  of first parameters of the LTP filter with one coefficient is determined,
- for the coding of a second current subframe, a plurality of pairs  $(\lambda_s, (\beta_i)_s)$  of parameters of the LTP filter with several coefficients is determined on the basis of the suite of parameters of the first format  $(\lambda_e, \beta_e)$ , with:
  - 10 - a determination of an LTP delay  $\lambda_s$  corresponding preferably to that determined by the first coding format on a first subframe which most overlaps the second current subframe,
  - a determination of a vector of gains  $(\beta_i)_s$  for the second current subframe on the basis of one at least of the gains  $\beta_e$  of the first subframes, by implementing steps b), c) and d) where the orders of the dictionary of the second format correspond to a set of gain vectors  $(\beta_i)_s$  of the second subframe.

12. Method according to Claim 11, **characterized in that**, for the coding of a second current subframe:

- 15 - on the basis of first LTP gains of the first format  $(\beta_e)$  that are chosen for one or more first subframes corresponding to a second current subframe, the orders of the dictionary of the second format, that are associated with classes of the first LTP gains, are preselected,
- 20 - a single one of these orders is constructed, preferably dynamically, on the basis of said orders preselected for said second current subframe, and
- N first vectors of seconds gains, determined by the order constructed, are tested so as to select, according to a chosen criterion, a better vector of gains to be associated with the second subframe.

13. Method according to one of Claims 1 to 10, in which the second coding format uses a filtering with one coefficient for second LTP subframes while the first coding format uses a filtering with several coefficients for first LTP subframes, **characterized in that:**

- 25 - for each first subframe, by implementing the first coding format, a first suite of LTP parameters  $\lambda_e, (\beta_i)_e$  corresponding to a pair comprising an LTP delay  $\lambda_e$  and a vector of associated gains  $(\beta_i)_e$  of the LTP filter with several coefficients is determined,
- 30 - a partition of a dictionary of the gain vectors  $(\beta_i)_e$  of the first format is performed,
- for the coding of a second current subframe by the second format, orders of a dictionary of the second format are determined for first subframes corresponding to the second current subframe, said dictionary of the second format being constructed from a set of jitter values and said orders of this dictionary being associated with the partition of the dictionary of the first format,
- 35 - an order of the jitter values is determined and LTP delay values for the second format are explored successively on the jitter values thus ordered and about one or more anchoring delays determined as a function of the delays  $\lambda_e$  on the first subframes.

14. Method according to one of Claims 1 to 10, in which the first coding format uses a filtering with several coefficients on first LTP subframes and the second coding format uses a filtering with several coefficients on second LTP subframes, **characterized in that:**

- 45 - on the basis of at least one first suite of parameters selected by the first format and including at least one vector of gains  $(\beta_i)_e$  determined for at least one first subframe, a partition is conducted of the dictionary of the first format corresponding to a dictionary of the gain vectors of the first format  $(\beta_i)_e$ ,
- orders of the dictionary of the second format corresponding to a dictionary of the gain vectors  $(\beta_i)_s$  of the second format are deduced therefrom, said orders being associated with said partition,
- 50 - on the basis of gain vectors  $(\beta_i)_e$  chosen by the first format for first subframes which substantially cover the second current subframe, orders of the second dictionary that are associated with classes of said partition are preselected,
- one of the preselected orders is retained,
- several gain vectors to be associated with the second current subframe are determined as a function of the order retained,
- 55 - by tests on said several gain vectors, the best gain vector is selected according to a chosen criterion.

15. Device for coding an audio signal according to a second format, designed to use coding information obtained by

implementing a coding according to a first format, the first and second formats implementing, in particular for the coding of a speech signal, a search for LTP long-term prediction parameters by exploring a dictionary comprising candidate parameters, one at least of the first and second coding formats using a filtering with several coefficients for a fine search for the LTP parameters, **characterized in that** it comprises:

- 5 - a memory storing a correspondence table defining, as a function of LTP parameters determined by the first coding format, orders of a dictionary that the second coding format uses, the correspondence table being defined on the basis of results of a statistical and/or analytical study, conducted as a function of successive sets of LTP parameters according to the first coding format, so as to determine a number of orders and appropriate orders in a dictionary used by the second coding format,
- 10 - means for recovering a signal giving at least one *a priori* partitioning information of the first dictionary relating to a class of the partition to which an LTP parameter obtained in the course of the coding according to the first format belongs, and obtained following the determination of the LTP parameters in the course of a coding according to the first format, so as to select at least one order of said dictionary that the second coding format used,
- 15 - means active on reception of said signal for consulting said correspondence table and selecting at least one order of said dictionary that the second coding format uses,
- calculation means for:
- ordering said dictionary that the second coding format uses according to the selected order, with a view to choosing a limited number of first candidates from the dictionary, and
- 20 - continuing the coding according to the second format, by conducting the LTP search only among this limited number of candidates.

16. Coding system implementing at least one first and one second coding format, **characterized in that** it comprises at least one device for coding according to the first format and a coding device according to Claim 15, applying said second format.

17. Computer program product, stored in a memory of a processing unit or on a removable medium intended to cooperate with a reader of said processing unit or downloadable from a remote site, **characterized in that** it comprises instructions for implementing all or some of the steps of the method according to one of Claims 1 to 14.

### Patentansprüche

1. Verfahren zum Codieren eines Audiosignals in einem zweiten Format anhand von Informationen, die durch Ausführen wenigstens eines Schrittes zum Codieren in einem ersten Format erhalten werden, wobei das erste und das zweite Format, insbesondere zum Codieren eines Sprachsignals, einen Schritt zum Suchen von LTP-Parametern zur langfristigen Vorhersage durch Untersuchen wenigstens eines Wörterbuchs, das Kandidatenparameter enthält, ausführen, wobei das erste und/oder das zweite Codierungsformat eine Filterung mit mehreren Koeffizienten für eine feine Suche von LTP-Parametern verwenden, **dadurch gekennzeichnet, dass** es die folgenden Schritte umfasst:

- 45 - Zugreifen auf Ergebnisse einer statistischen und/oder analytischen Untersuchung, die als Funktion aufeinander folgender Sätze von LTP-Parametern in dem ersten Codierungsformat ausgeführt werden, um eine Anzahl von Befehlen und geeignete Befehle in einem Wörterbuch, das das zweite Codierungsformat verwendet, zu bestimmen,
- Wiedergewinnen einer hypothetischen Information für die Partition des ersten Wörterbuchs, die eine Klasse der Partition betrifft, zu der ein während der Codierung in dem ersten Format erhaltener LTP-Parameter gehört und die nach der Bestimmung der LTP-Parameter während der Codierung in dem ersten Format erhalten wird, um wenigstens einen Befehl des Wörterbuchs zu wählen, das das zweite Codierungsformat verwendet,
- 50 - Anwenden des gewählten Befehls auf Kandidaten des Wörterbuchs, das das zweite Codierungsformat verwendet, um eine begrenzte Anzahl erster Kandidaten zu wählen, und
- zum Ausführen der zweiten Codierung Ausführen der LTP-Suche ausschließlich in der begrenzten Anzahl von Kandidaten.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet, dass** zunächst eine elementare Partition des ersten Wörterbuchs vorgesehen wird, die N Elemente in N disjunkten Klassen mit der Größe 1 enthält.

3. Verfahren nach Anspruch 1, wobei das erste Codierungsformat ein erstes Wörterbuch verwendet und das zweite

Codierungsformat ein zweites Wörterbuch verwendet, **dadurch gekennzeichnet, dass** eine Partition des ersten Wörterbuchs in nicht disjunkten Klassen vorgesehen wird, derart, dass dasselbe Element mehr als einem Befehl des zweiten Wörterbuchs zugeordnet sein kann.

- 5     **4.** Verfahren nach einem der Ansprüche 1 bis 3, wobei das erste Codierungsformat ein erstes Wörterbuch verwendet und das zweite Codierungsformat ein zweites Wörterbuch verwendet, **dadurch gekennzeichnet, dass** eine Umgruppierung ähnlicher Befehle vorgesehen ist, um die anfängliche Partition des ersten Wörterbuchs und von hier aus die Anzahl von Befehlen des zweiten Wörterbuchs dynamisch zu modifizieren.
- 10    **5.** Verfahren nach Anspruch 4, **dadurch gekennzeichnet, dass** außerdem eine Operation vorgesehen ist, die darin besteht, die Befehle des zweiten Wörterbuchs nacheinander neu zu berechnen, sobald sie umgruppiert worden sind, und dass die anfängliche Partition des ersten Wörterbuchs und/oder die auf diese Weise umgruppierten Befehle dynamisch modifiziert werden.
- 15    **6.** Verfahren nach einem der Ansprüche 4 bis 5, wobei das erste Codierungsformat ein erstes Wörterbuch verwendet und das zweite Codierungsformat ein zweites Wörterbuch verwendet, **dadurch gekennzeichnet, dass** für jeden der Befehle des zweiten Wörterbuchs eine zu berücksichtigende maximale Anzahl von Elementen des zweiten Wörterbuchs als Funktion der Klassen des ersten Wörterbuchs und/oder der Befehle des zweiten Wörterbuchs gewählt wird, um ein für die Speicherung der Befehle des zweiten Wörterbuchs verwendetes Speicherbetriebsmittel zu begrenzen.
- 20    **7.** Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet, dass** die begrenzte Anzahl von Kandidaten als Funktion eines Kompromisses zwischen der Qualität und der Komplexität der zweiten Codierung gewählt wird.
- 25    **8.** Verfahren nach Anspruch 7, wobei ein zu codierendes Eingangssignal in Datenblöcken verarbeitet wird, **dadurch gekennzeichnet, dass** der Kompromiss bei jedem zu verarbeitenden Datenblock als Funktion von Parametern des ersten Codierungsformats und/oder von Charakteristiken des zu codierenden Signals und vorzugsweise als Funktion von LTP-Unterrahmen, die jeder Datenblock enthält, festgelegt wird.
- 30    **9.** Verfahren nach einem der Ansprüche 1 bis 8, wobei ein zu codierendes Eingangssignal in Datenblöcken verarbeitet wird, wovon jeder für das erste Codierungsformat erste LTP-Unterrahmen enthält und für das zweite Codierungsformat zweite LTP-Unterrahmen enthält, **dadurch gekennzeichnet, dass** für die ersten und zweiten Unterrahmen mit gleicher Dauer jedem momentanen Unterrahmen des zweiten Codierungsformats ein einziger Unterrahmen des ersten Codierungsformats entspricht, und dass:
- 35    - das erste Codierungsformat einen ersten Satz von LTP-Parametern für den momentanen Unterrahmen wählt,  
 - anhand der Partition nach Klassen des Wörterbuchs, die einem der LTP-Parameter des ersten Formats zugeordnet ist, ein Befehl zum Auswerten des Wörterbuchs des zweiten Formats gewählt wird, indem ein der Klasse des Elements des ersten Satzes zugeordneter Befehl gewählt wird, und  
 - gemäß dem auf diese Weise gewählten Befehl eine begrenzte Anzahl erster Kandidaten des Wörterbuchs des zweiten Formats untersucht wird.
- 40    **10.** Verfahren nach einem der Ansprüche 1 bis 8, wobei ein zu codierendes Eingangssignal in Datenblöcken verarbeitet wird, wovon jeder für das erste Codierungsformat erste LTP-Unterrahmen und für das zweite Codierungsformat zweite LTP-Unterrahmen enthält, **dadurch gekennzeichnet, dass** für erste und zweite Unterrahmen mit unterschiedlicher Dauer:
- 45    - das erste Codierungsformat mehrere Sätze von LTP-Parametern für erste Unterrahmen, die im Wesentlichen einem momentanen zweiten Unterrahmen entsprechen, wählt,  
 - anhand der Partition nach Klassen des Wörterbuchs, die einem der LTP-Parameter des ersten Formats zugeordnet ist, im Voraus Befehle zum Untersuchen des Wörterbuchs des zweiten Formats gewählt werden, indem die den Klassen der Elemente der Sätze von LTP-Parametern zugeordneten Befehle gewählt werden,  
 - anhand der Vorauswahl dieser Befehle wenigstens ein bevorzugter Befehl bestimmt wird und  
 - das Wörterbuch des zweiten Formats gemäß dem bevorzugten Befehl untersucht wird, indem eine Beschränkung auf seine ersten Elemente erfolgt.
- 50    **11.** Verfahren nach einem der vorhergehenden Ansprüche, wobei das erste Codierungsformat eine Filterung mit einem
- 55

einigen Koeffizienten für die ersten LTP-Unterrahmen verwendet, während das zweite Codierungsformat eine Filterung mit mehreren Koeffizienten für die zweiten LTP-Unterrahmen verwendet, **dadurch gekennzeichnet, dass:**

- für jeden ersten Unterrahmen durch Verwenden des ersten Codierungsformats ein Paar erster Parameter ( $\lambda_e$ ,  $\beta_e$ ) des LTP-Filters mit einem einzigen Koeffizienten bestimmt wird,
- für die Codierung eines momentanen zweiten Unterrahmens mehrere Paare von Parametern ( $\lambda_s$ ,  $(\beta_i)_s$ ) des LTP-Filters mit mehreren Koeffizienten auf der Grundlage des Satzes von Parametern ( $\lambda_e$ ,  $\beta_e$ ) des ersten Formats bestimmt werden, mit:

einer Bestimmung einer LTP-Verzögerung ( $\lambda_s$ ), die vorzugsweise jener entspricht, die durch das erste Codierungsformat an einem ersten Unterrahmen bestimmt wird, die den momentanen zweiten Unterrahmen am weitesten abdeckt,  
 einer Bestimmung eines Vektors von Verstärkungen  $(\beta_i)_s$  für den momentanen zweiten Unterrahmen anhand wenigstens einer Verstärkung  $\beta_e$  der ersten Unterrahmen für die Ausführung der Schritte b), c) und d), wobei die Befehle des Wörterbuchs des zweiten Formats einer Gesamtheit von Verstärkungsvektoren  $(\beta_i)_s$  des zweiten Unterrahmens entsprechen.

12. Verfahren nach Anspruch 11, **dadurch gekennzeichnet, dass** für die Codierung eines zweiten momentanen Unterrahmens:

- anhand erster LTP-Verstärkungen des ersten Formats ( $\beta_e$ ), die für einen oder mehrere erste Unterrahmen gewählt werden, die einem zweiten momentanen Unterrahmen entsprechen, die Befehle des Wörterbuchs des zweiten Formats, die Klassen erster LTP-Verstärkungen zugeordnet sind, im Voraus gewählt werden,
- ein einziger dieser Befehle vorzugsweise dynamisch anhand der im Voraus gewählten Befehle für den zweiten momentanen Unterrahmen gebildet wird, und
- N erste Vektoren von zweiten Verstärkungen, die durch den gebildeten Befehl bestimmt werden, getestet werden, um gemäß einem gewählten Kriterium einen besten Verstärkungsvektor zu wählen, der dem zweiten Unterrahmen zugeordnet werden soll.

13. Verfahren nach einem der Ansprüche 1 bis 10, wobei das zweite Codierungsformat eine Filterung mit einem einzigen Koeffizienten für zweite LTP-Unterrahmen verwendet, während das erste Codierungsformat eine Filterung mit mehreren Koeffizienten für erste LTP-Unterrahmen verwendet, **dadurch gekennzeichnet, dass:**

- für jeden ersten Unterrahmen für die Verwendung des ersten Codierungsformats ein erster Satz von LTP-Parametern  $\lambda_e$ ,  $(\beta_i)_e$  bestimmt wird, der einem Paar entspricht, das eine LTP-Verzögerung  $\lambda_e$  und einen Vektor  $(\beta_i)_e$  zugeordneter Verstärkungen des LTP-Filters mit mehreren Koeffizienten enthält,
- eine Partition eines Wörterbuchs von Verstärkungsvektoren  $(\beta_i)_e$  des ersten Formats ausgeführt wird,
- für die Codierung eines zweiten momentanen Unterrahmens mit dem zweiten Format Befehle eines Wörterbuchs des zweiten Formats für erste Unterrahmen, die dem zweiten momentanen Unterrahmen entsprechen, bestimmt werden, wobei das Wörterbuch des zweiten Formats aus einer Gesamtheit von Jitter-Werten gebildet ist und die Befehle dieses Wörterbuchs der Partition des Wörterbuchs des ersten Formats zugeordnet sind,
- ein Befehl der Jitter-Werte bestimmt wird und nacheinander Werte von LTP-Verzögerungen für das zweite Format an den Jitter-Werten, die auf diese Weise geordnet worden sind, und um eine oder mehrere Verankerungsverzögerungen, die als Funktion der Verzögerungen  $\lambda_e$  in den ersten Unterrahmen bestimmt werden, untersucht werden.

14. Verfahren nach einem der Ansprüche 1 bis 10, wobei das erste Codierungsformat eine Filterung mit mehreren Koeffizienten an ersten LTP-Unterrahmen verwendet und das zweite Codierungsformat eine Filterung mit mehreren Koeffizienten an zweiten LTP-Unterrahmen verwendet, **dadurch gekennzeichnet, dass:**

- anhand wenigstens eines ersten Satzes von Parametern, der durch das erste Format gewählt wird und wenigstens einem Vektor von Verstärkungen  $(\beta_i)_e$  enthält, der für wenigstens einen ersten Unterrahmen bestimmt wird, eine Partition des Wörterbuchs des ersten Formats vorgenommen wird, die einem Wörterbuch von Verstärkungsvektoren des ersten Formats  $(\beta_i)_e$  entspricht,
- daraus Befehle des Wörterbuchs des zweiten Formats abgeleitet werden, die einem Wörterbuch von Verstärkungsvektoren  $(\beta_i)_s$  des zweiten Formats entsprechen, wobei diese Befehle der Partition zugeordnet sind,
- anhand der Verstärkungsvektoren  $(\beta_i)_e$ , die mit dem ersten Format für erste Unterrahmen gewählt werden, die im Wesentlichen den momentanen zweiten Unterrahmen abdecken, im Voraus Befehle des zweiten Wör-



terbuchs gewählt werden, die Klassen dieser Partition zugeordnet sind,

- einer der im Voraus gewählten Befehle berücksichtigt wird,

- mehrere Verstärkungsvektoren, die dem zweiten momentanen Unterrahmen zugeordnet werden sollen, als Funktion des berücksichtigten Befehls bestimmt werden und

- durch Tests an den mehreren Verstärkungsvektoren der beste Verstärkungsvektor gemäß einem gewählten Kriterium ausgewählt wird.

15. Vorrichtung zum Codieren eines Audiosignals mit einem zweiten Format, die dazu ausgelegt ist, erhaltene Codierungsinformationen für die Ausführung einer Codierung mit einem ersten Format zu verwenden, wobei das erste und das zweite Format, insbesondere für das Codieren eines Sprachsignals, eine Suche von LTP-Parametern zur langfristigen Vorhersage durch Untersuchen eines Wörterbuchs, das Kandidatenparameter enthält, ausführen, wobei das erste und/oder das zweite Codierungsformat eine Filterung mit mehreren Koeffizienten für eine feine Suche von LTP-Parametern verwenden, **dadurch gekennzeichnet, dass** sie umfasst:

- einen Speicher, der eine Korrespondenztabelle speichert, die als Funktion von LTP-Parametern, die durch das erste Codierungsformat bestimmt werden, Befehle eines Wörterbuchs definiert, das das zweite Codierungsformat verwendet, wobei die Korrespondenztabelle anhand von Ergebnissen einer statistischen und/oder analytischen Untersuchung definiert ist, die als Funktion von aufeinander folgenden Sätzen von LTP-Parametern mit dem ersten Codierungsformat vorgenommen wird, um eine Anzahl von Befehlen und geeignete Befehle in einem Wörterbuch, das das zweite Codierungsformat verwendet, zu bestimmen,

- Mittel, um ein Signal wiederzugewinnen, das wenigstens eine hypothetische Information für die Partition des ersten Wörterbuchs angibt, die eine Klasse der Partition betrifft, zu der ein LTP-Parameter gehört, der während der Codierung mit dem ersten Format erhalten wird, und die nach der Bestimmung von LTP-Parametern während einer Codierung mit dem ersten Format erhalten wird, um wenigstens einen Befehl des Wörterbuchs zu wählen, das das zweite Codierungsformat verwendet,

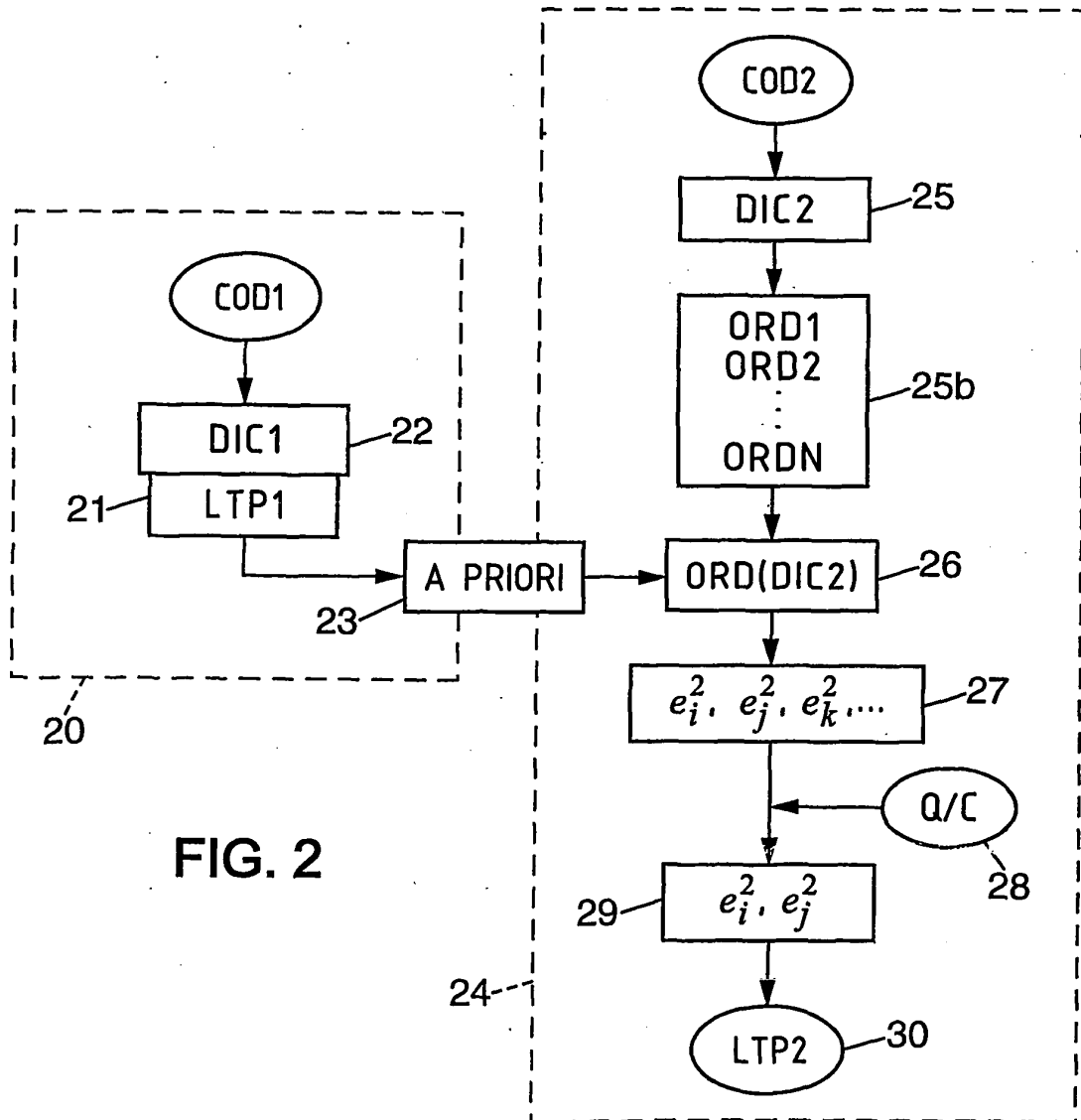
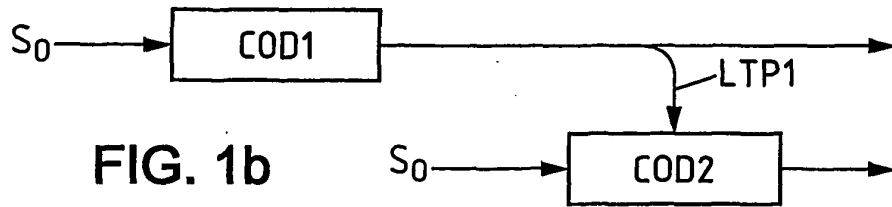
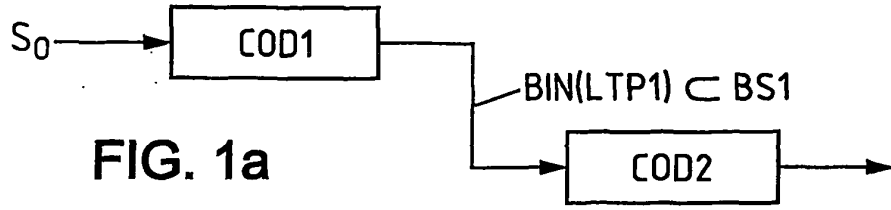
- aktive Mittel zum Empfangen des Signals, um die Korrespondenztabelle abzufragen und um wenigstens einen Befehl des Wörterbuchs, das das zweite Codierungsformat verwendet, zu wählen,

- Rechenmittel, um:

das Wörterbuch, das das zweite Codierungsformat verwendet, gemäß dem gewählten Befehl zu ordnen, um eine begrenzte Anzahl erster Kandidaten in dem Wörterbuch zu wählen, und die Codierung mit dem zweiten Format fortzusetzen, indem die LTP-Suche ausschließlich in dieser begrenzten Anzahl von Kandidaten vorgenommen wird.

16. Codierungsverfahren, das wenigstens ein erstes und ein zweites Codierungsformat verwendet, **dadurch gekennzeichnet, dass** es wenigstens eine Vorrichtung für die Codierung mit dem ersten Format und eine Codierungsvorrichtung nach Anspruch 15, die das zweite Format anwendet, umfasst.

17. Computerprogrammprodukt, das in einem Speicher einer Verarbeitungseinheit oder in einem entnehmbaren Träger, der dazu vorgesehen ist, mit einem Lesegerät der Verarbeitungseinheit zusammenzuwirken, gespeichert ist oder das von einem entfernten Ort fernladbar ist, **dadurch gekennzeichnet, dass** es Befehle für die Ausführung aller oder einiger Schritte des Verfahrens nach einem der Ansprüche 1 bis 14 enthält.



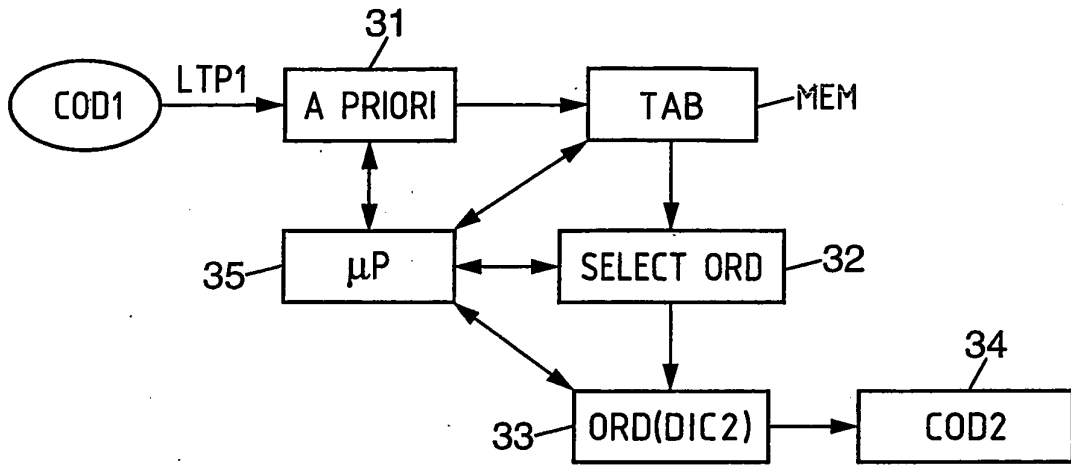


FIG. 3

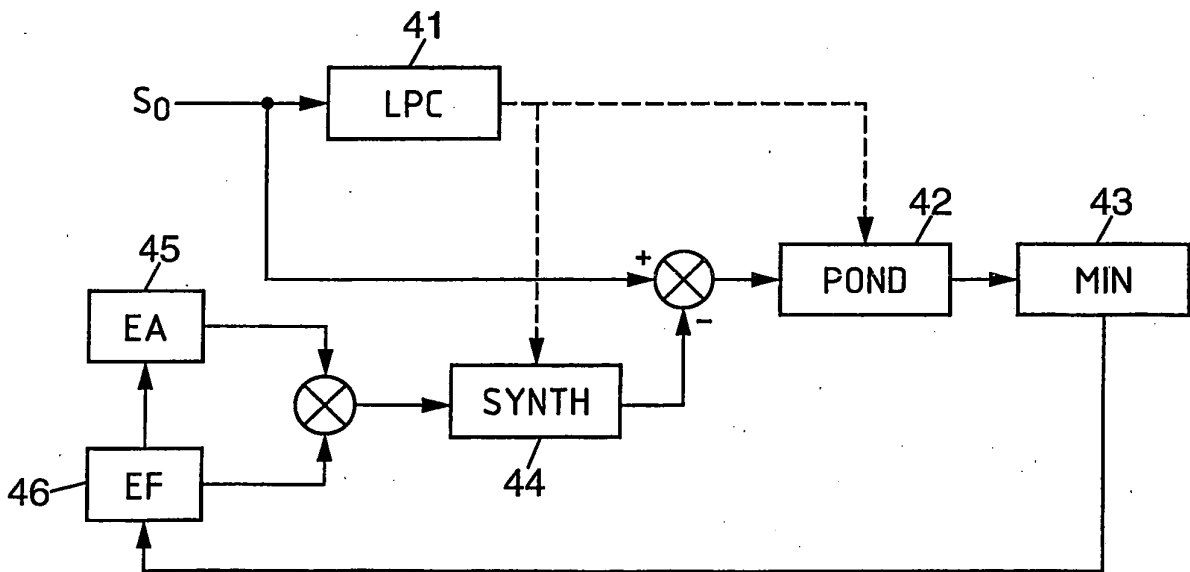
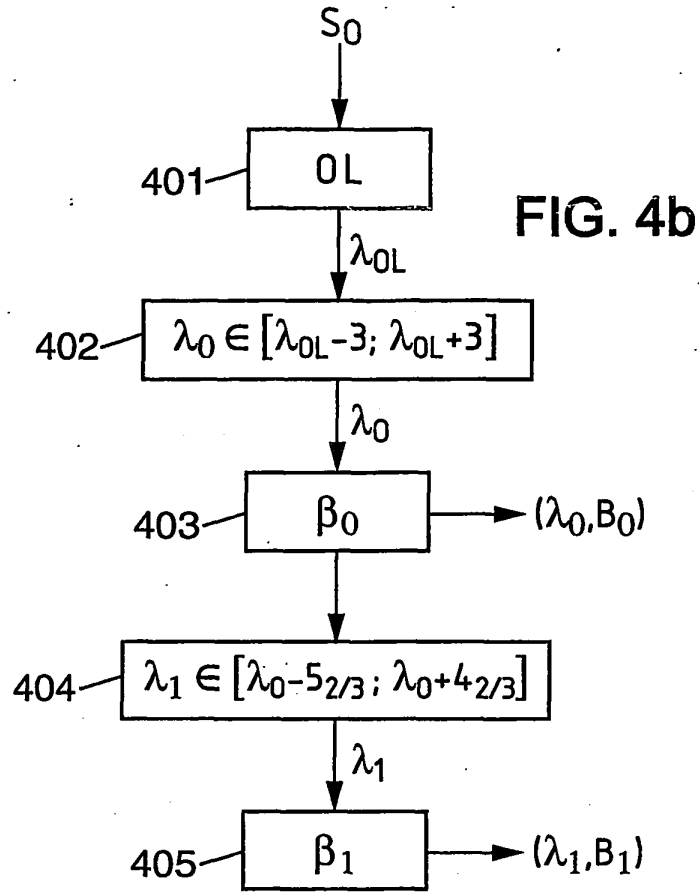
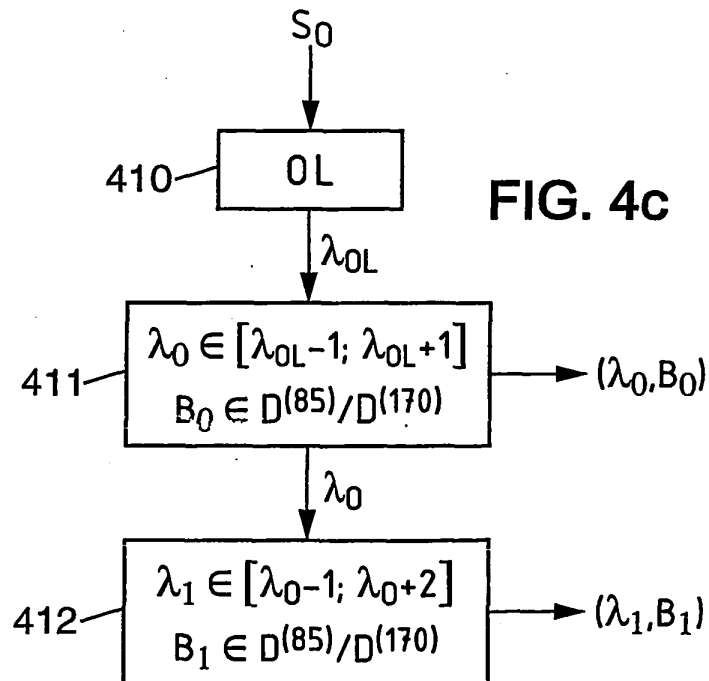


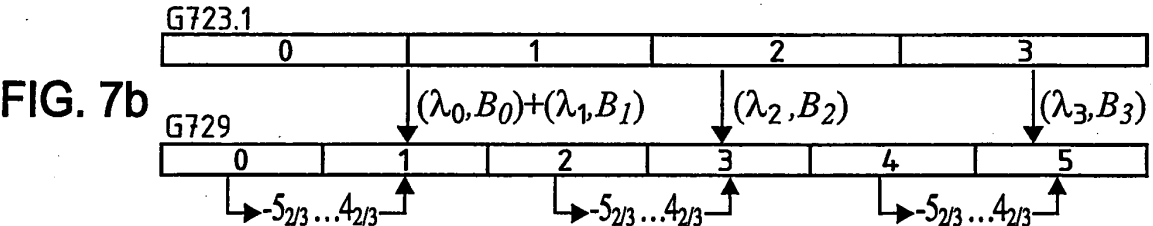
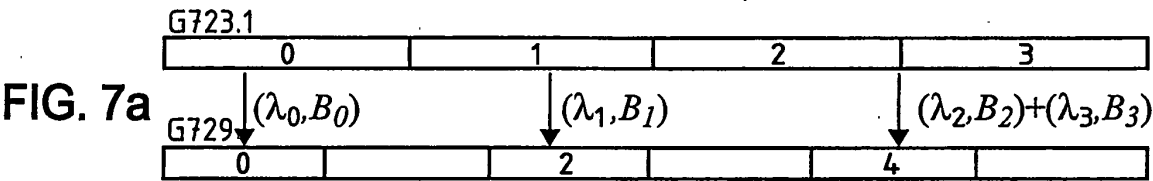
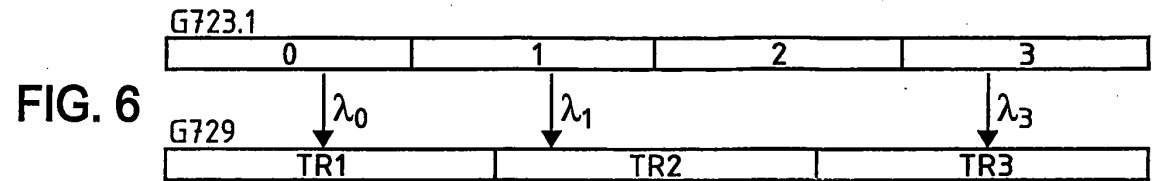
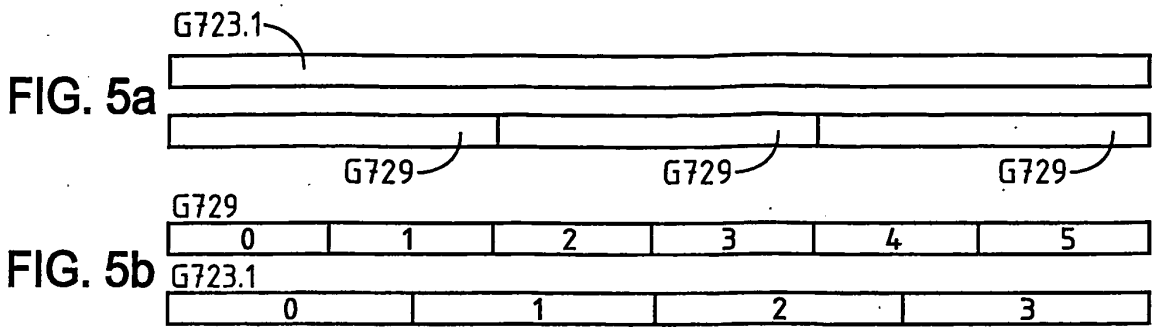
FIG. 4a



**FIG. 4b**



**FIG. 4c**



CG	G729	G723.1	CD
	0	0	
	1	0,1	
	2	1	
	3	2	
	4	2,3	
	5	3	

**FIG. 8**

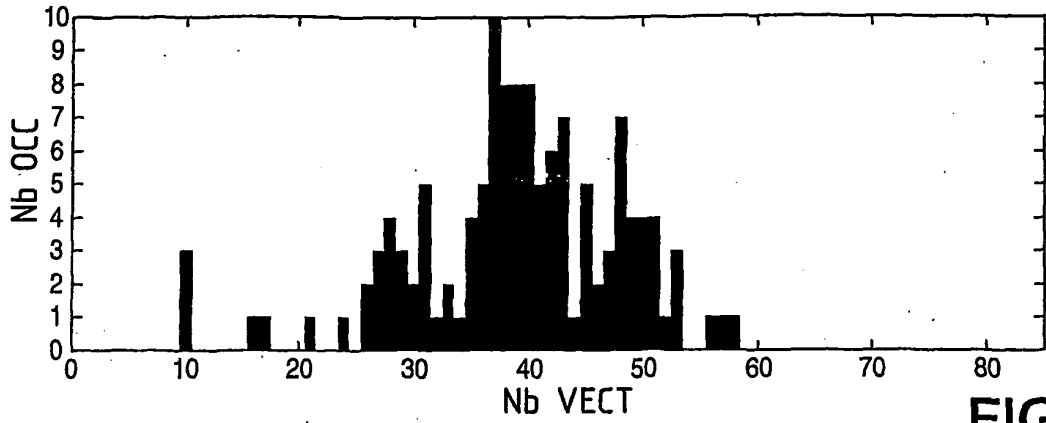


FIG. 9a

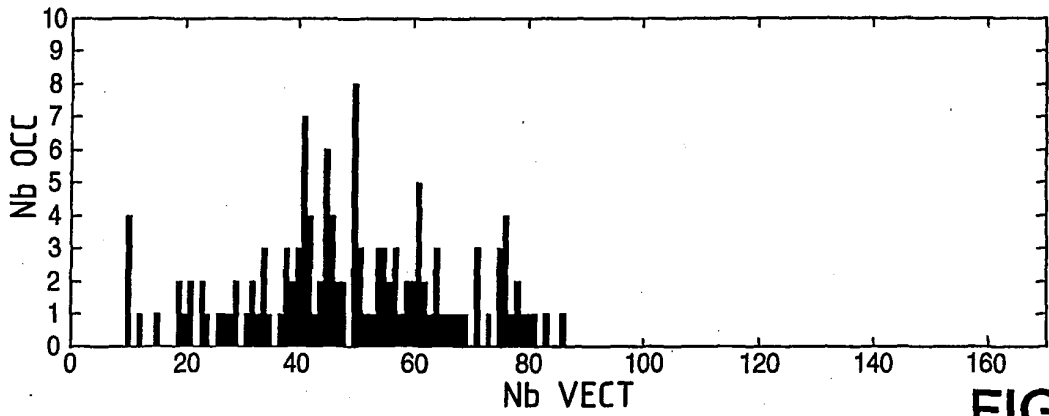


FIG. 9b

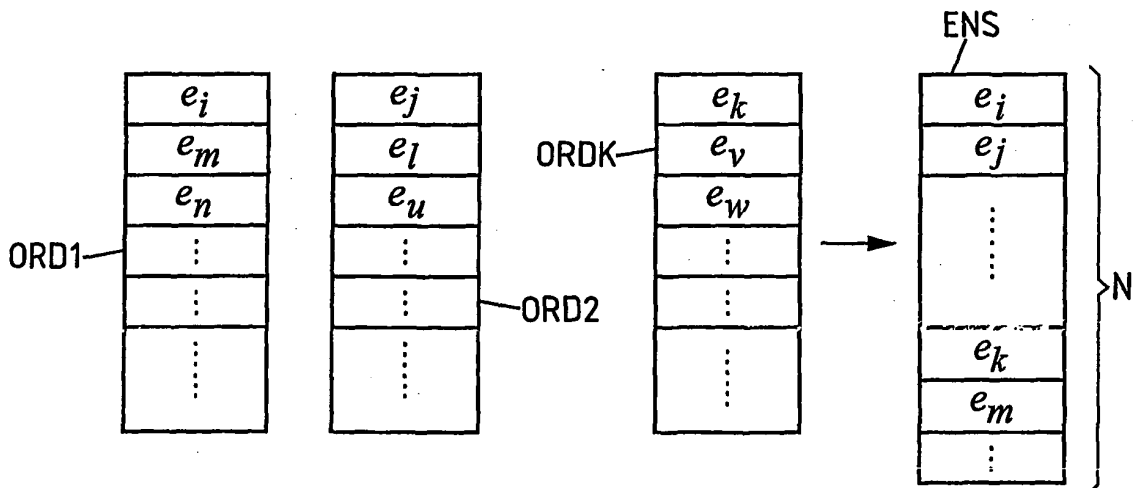


FIG. 10

**RÉFÉRENCES CITÉES DANS LA DESCRIPTION**

*Cette liste de références citées par le demandeur vise uniquement à aider le lecteur et ne fait pas partie du document de brevet européen. Même si le plus grand soin a été accordé à sa conception, des erreurs ou des omissions ne peuvent être exclues et l'OEB décline toute responsabilité à cet égard.*

**Documents brevets cités dans la description**

- WO 03058407 A [0015] [0019] [0059]

**Littérature non-brevet citée dans la description**

- **Sung-Wan Yoon ; Sung-Kyo Jung ; Young-Cheol Park ; Dae-Hee Youn.** An Efficient Transcoding Algorithm For G. 723.1 And G. 729A Speech Coders. *Proc. Eurospeech*, 2001, 2499-2502 [0015]
- **Sung-Wan Yoon.** Transcoding algorithm for G723.1 and AMR speech Coders: for interoperability between VoIP and Mobile Networks. *Proc. Eurospeech*, 2003, 1101-1104 [0015]