الجمهورية الجزائرية الديمقراطيــة الشعبيــة

République Algérienne Démocratique et Populaire

وزارة التعليـم العالـي والبحـث العلمـي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des frères Mentouri Constantine 1 جامعة الاخوة منتوري قسنطينة 1

Faculté des Sciences de la Technologie كليــة علوم التكنولوجية

Département d'Electronique قسم الإلكترونيك

Laboratoire Signaux et Systèmes de Communication (SISCOM)

N° d'Ordre: 55/D3C/2021 **Série: 03/Elec/2021**

Thèse :

Présentée pour Obtenir le Diplôme de

Doctorat Troisième Cycle

Filière : Télécommunications

Spécialité : Signaux et Systèmes de Télécommunications

Techniques de Détection de l'Activité Vocale dans un Canal de Communication : Comparaison aux Standards de Compression Audio

Par : Chelloug Charaf Eddine

Présentée et soutenue publiquement devant le jury :

Président	**Faouzi Soltani**	**Professeur Université des Frères Mentouri de Constantine 1**
Rapporteur	**Farrouki Atef**	**Professeur Université des Frères Mentouri de Constantine 1**
Examinateur	**Fella Hachouf**	**Professeur Université des Frères Mentouri de Constantine 1**
Examinateur	**Karim Kemih**	**Professeur Université de Jijel**
Examinateur	**Fortaki Tarek**	**Professeur Université de Batna 2**

16/06/2021

Année Universitaire 2020/2021

Dédicaces

Je dédie ce mémoire

A la mémoire de mon inoubliable père qui m'a tant donné et qui aurait aimé voir ce jour et partager ma joie. Je ne t'oublierai jamais, papa.
Qu'il repose en paix.

A ma mère, dont l'incommensurable contribution à mon éducation, à mon instruction et à tous les instants de ma vie, ravivera jusqu'à la fin de mes jours mon infinie tendresse. Qu'elle trouve ici l'expression de mon éternelle reconnaissance.

A mon grand frère Abd Raouf, sa femme et ses enfants Amjed et Abderrahmene

A mon petit frère Anis, sa femme et sa petite ange Asil

A toute ma famille.

A tous mes amis et amies.

A tous mes collègues, mes enseignants et personnel administratif, toute ma reconnaissance et mon attachement.

A toutes les personnes que j'aime et que j'ai involontairement oubliées.

Remerciements

Le travail présenté dans cette thèse de Doctorat a été élaboré au sein du Laboratoire Signaux et Systèmes de Communication (SISCOM), Département d'électronique, Université des Frères Mentouri Constantine 1

Mes remerciements vont en premier lieu à dieu le tout puissant, pour m'avoir accordé la santé, la volonté et le courage.

Je remercie

Mon directeur de thèse

Monsieur Le Professeur Farrouki Atef

Je vous remercie de m'avoir honoré avec votre confiance et de m'avoir confié un excellent sujet de recherche

Je vous remercie également pour votre sérieux, votre temps et vos conseils fructueux

Monsieur le Président de Jury :

Le Professeur Soltani Faouzi

Je vous remercie pour l'honneur que vous me faites en acceptant d'évaluer ce travail

Veuillez Monsieur trouver ici, l'assurance de ma reconnaissance et de ma profonde admiration

Mes vifs remerciements aux membres du jury, Les Professeurs :

- ✓ ***Fella Hachouf***
- ✓ ***Kemih Karim***
- ✓ ***Fortaki Tarek***

Je tiens également à remercier

Tous les professeurs du Laboratoire SISCOM pour leurs conseils, leur gentillesse et leur sérieux.

Enfin, je remercie tous les chercheurs dans le domaine de la détection d'activité vocale

Ainsi que dans tous les domaines de télécommunication pour l'intérêt qu'ils portent à leur travail

Résumé

La détection d'activité vocale VAD (Voice Activity Detection) consiste à déterminer les segments sonores avec et sans parole dans une conversation. Cette étape est cruciale dans la plupart des systèmes de traitement de la parole car elle permet d'identifier et de délimiter avec précision l'information utile; cette opération est relativement aisée dans un environnement faiblement perturbé, par contre dans le cas d'un bruit important, il devient difficile de distinguer entre les régions du silence et celles d'activité vocale.

Les méthodes de détection d'activité vocale sont utilisées dans de nombreux domaines, notamment dans le codage de la parole, dans le rehaussement de la parole (speech enhancement) ou encore dans la reconnaissance vocale. En outre, ces techniques sont présentes dans diverses applications telles que les services de communication mobile, la voix sur IP (Internet Protocol) ou la transmission vocale dans les systèmes d'aide auditive.

A titre indicatif, un VAD réalise la compression du silence dans les systèmes de télécommunications mobiles modernes en réduisant le débit moyen via le mode de transmission discontinue (DTX Discontinuous Transmission). Beaucoup de systèmes de communication, tels que la téléphonie mobile (2G, 3G, 4G), utilisent la détection des intervalles de silence, associée à un mode de transmission discontinue, pour une efficacité de codage et de compression plus accrue.

Dans cette thèse, nous décrivons les principales méthodes VAD normalisées et mentionnées dans la littérature; telles que le standard VAD G.729-B approuvé par l'UIT-T en 1996, le VAD AMR (Adaptatif Multi Rate), le VAD AFE (Advanced front-end) et le SILK (développé par Skype).
Le VAD du standard G.729, appelé communément annexe B au niveau de la norme, génère une décision VAD pour chaque trame en fonction de quatre paramètres pertinents extraits du signal audio. D'une manière simplifiée, ces paramètres sont directement liés à l'énergie et au contenu spectral d'une trame vocale. Le G-729-B est utilisé dans la majorité des applications de transmission audio, au point de devenir la technique VAD conventionnelle la plus populaire. Par conséquent, le G-729-B est aujourd'hui la référence par excellence pour toute étude comparative dans la plupart des articles scientifiques traitant les VAD.

Dans la 1ère contribution, nous proposons un schéma VAD basé sur un seuillage adaptatif tout en maintenant le taux de fausse acceptation (False Acceptance Rate) à une valeur nominale. Comme admis dans la théorie de la décision binaire, le taux d'erreur, noté « False Acceptance Rate », s'apparente à la probabilité de classer une trame de silence comme Voix Active. L'idée de base consiste à mettre en œuvre des tests séquentiels, basés sur l'énergie pleine bande, afin de rejeter ou d'accepter la trame en cours d'investigation en tant que région vocale active. La caractéristique la plus intéressante de l'algorithme proposé réside dans sa capacité à mettre à jour dynamiquement l'estimateur du niveau de bruit en fonction de l'environnement en cours. En tenant compte de la stationnarité à long terme du signal parole, nous avons également développé une procédure de lissage (Smoothing) afin de minimiser les discontinuités pouvant avoir lieu dans les régions vocales et non vocales.
Les performances de l'approche proposée ont été évaluées et comparées au VAD de la norme G.729-B dans plusieurs situations incluant divers bruits acoustiques environnementaux avec différents SNR. L'analyse des résultats a été réalisée en utilisant la base de données expérimentale NOIZEUS ainsi que des signaux vocaux réels enregistrés.

La 2ème contribution consiste à implémenter l'approche proposée sur un système à base de microcontrôleur, afin de :

- *S'assurer de la robustesse de l'algorithme,*
- *Evaluer sa complexité d'implémentation*
- *S'assurer de son fonctionnement en temps réel*

Dans ce contexte, divers tests on été conduits en temps réel à l'aide d'outils d'analyse et de développement accompagnant le système à microcontrôleur (STM32F7). Ces outils nous ont permis d'effectuer un monitoring en temps réel dans des situations réalistes, en visualisant le signal en cours d'enregistrement, le seuil adaptatif par rapport à l'énergie en pleine bande ainsi que la décision VAD pour chaque trame traitée. Grace à cette façon de procéder, nous avons pu déterminer avec précision le temps de traitement (Latence) nécessaire à la génération d'une décision finale pour chaque trame. L'analyse en temps réel de notre système a permis d'obtenir une latence de 4 µs, ce qui semble amplement suffisant pour garantir un fonctionnement en temps réel en tenant compte des fréquences d'échantillonnage les plus couramment utilisées en traitement de la parole (8 kHz à 16 kHz).

ملخص

يعتمد اكتشاف النشاط الصوتي (VAD Voice Activity Detection) على تحديد مقاطع الصوت مع وبدون كلام في المحادثة. هذه الخطوة حاسمة في معظم أنظمة معالجة الكلام لأنها تسمح بتحديد المعلومات المفيدة وتحديدها بدقة ؛تعتبر هذه العملية سهلة نسبيًا في بيئة مضطربة بشكل ضعيف، ومن ناحية أخرى ، في حالة الضوضاء الكبيرة ، يصبح من الصعب التمييز بين مناطق الصمت ومناطق النشاط الصوتي.

تُستخدم طرق اكتشاف النشاط الصوتي في العديد من المجالات ، لا سيما في ترميز الكلام أو تحسين الكلام أو في التعرف على الكلام. بالإضافة إلى ذلك ، توجد هذه التقنيات في تطبيقات مختلفة مثل خدمات الاتصالات المتنقلة أو نقل الصوت عبر بروتوكول الإنترنت IP (Internet Protocol) أو الإرسال الصوتي في أنظمة المعينات السمعية.

كمؤشر ، يحقق VAD ضغط الصمت في أنظمة الاتصالات المتنقلة الحديثة عن طريق تقليل متوسط معدل البت عبر وضع الإرسال غير المستمر (DTX Discontinuous Transmission). تستخدم العديد من أنظمة الاتصالات ، مثل الهاتف المحمول (2G، 3G، 4G) كشف الفاصل الزمني الصامت ، جنبًا إلى جنب مع وضع الإرسال غير المستمر ، لزيادة كفاءة الترميز والضغط.

في هذه الأطروحة، نصف الطرق النموذجية الرئيسية لاكتشاف النشاط الصوتي الاكثر استعمالا و تداولا في البحوث العلمية ؛ مثل تقنية VAD G.729-B المعتمد من قبل قطاع (UIT-T) في عام 1996 ، و VAD AMR (Adaptif Multi Rate) ، و VAD AFE (Advanced Front End) و SILK (تم تطويره بواسطة Skype).
يُنشئ VAD G.729 ، الذي يشار إليه عادةً باسم الملحق B ، قرار VAD لكل اطار زمني اعتمادا على اربع خصائص أساسية مستخرجة من الإشارة الصوتية. بطريقة مبسطة ، ترتبط هذه المعلمات ارتباطًا مباشرًا بالطاقة والمحتوى الطيفي لإطار صوتي. يتم استخدام G-729-B في معظم تطبيقات نقل الصوت والتي أصبحت أكثر تقنيات VAD التقليدية شيوعًا. لذلك ، فإن G-729-B هي اليوم المعيار الممتاز لأي دراسة استنادا لمعظم المقالات العلمية التي تتناول VAD.

في المساهمة الأولى ، نقترح مخطط VAD على أساس العتبة التكيفية مع الحفاظ على معدل القبول الخاطئ بقيمة اسمية. كما هو متداول في نظرية القرار الثنائي ، فإن معدل الخطأ ، المشار إليه بـ "معدل القبول الخاطئ " ، مرتبط باحتمالية تصنيف إطار الصمت على أنه صوت نشط. الفكرة الأساسية هي إجراء اختبارات متسلسلة، بناءً على طاقة النطاق الكامل، من أجل رفض أو قبول الإطار الزمني قيد التحقيق كمنطقة صوت نشطة. تكمن الميزة الأكثر إثارة لاهتمام للخوارزمية المقترحة على قدرتها على التحديث الديناميكي المقدر لمستوى الضوضاء في للبيئة الحالية. مع الأخذ في الاعتبار الثبات طويل المدى لإشارة الكلام ، قمنا أيضًا بتطوير طريقة لتقليل الانقطاع التي يمكن أن تحدث في المناطق الصوتية وغير الصوتية.

تم تقييم أداء النهج المقترح ومقارنته بـ G.729-B VAD في العديد من المواقف بما في ذلك الضوضاء الصوتية البيئية المختلفة مع مستويات ضوضاء مختلفة. تم إجراء تحليل النتائج باستخدام قاعدة البيانات التجريبية NOIZEUS وكذلك الإشارات الصوتية المسجلة.

تتمثل المساهمة الثانية في تنفيذ النهج المقترح على نظام قائم على وحدة التحكم الدقيقة ، من أجل:

- ضمان متانة الخوارزمية
- تقييم مدى تعقيد التنفيذ
- التأكد من أنه يعمل في الوقت الحقيقي

في هذا السياق ، تم إجراء العديد من الاختبارات في الوقت الفعلي باستخدام أدوات التحليل والتطوير المصاحبة لنظام الميكروكونترولر (STM32F7). سمحت لنا هذه الأدوات بإجراء مراقبة في الوقت الفعلي في الحالات الواقعية ، من خلال تصور الإشارة التي يتم تسجيلها ، والعتبة التكيفية فيما يتعلق بطاقة النطاق الكامل وكذلك قرار VAD لكل إطار معالج. بفضل طريقة المتابعة هذه ، تمكنا من تحديد وقت المعالجة المطلوب لإنشاء قرار نهائي لكل إطار بدقة. سمح لنا التحليل في الوقت الفعلي لنظامنا بالحصول على زمن انتقال قدره 4 ميكروثانية ، والذي يبدو كافياً لضمان التشغيل في الوقت الفعلي مع مراعاة ترددات أخذ العينات الصوتية الأكثر شيوعًا في معالجة الكلام (8 كيلو هرتز إلى 16 كيلو هرتز).

Abstract

The main goal of the Voice Activity Detection (VAD) techniques is to distinguish between voiced regions and silent intervals in an audio communication. This step is crucial in most speech processing systems such as mobile communications, Voice over IP, speech recognition and hearing aid systems. This task seems to be relatively easy in a weakly disturbed environment. However, in the case of strongly noise, it becomes difficult to provide accurate information about the presence of active voice. In general, a VAD achieves the compression of silence intervals in modern communications systems by reducing the average bit rate via the discontinuous transmission mode (DTX).

In this thesis, we describe the main standardized VAD methods mentioned in the literature; namely the VAD G.729-B approved by ITU-T in 1996, the AMR VAD (Adaptive Multi Rate), the AFE VAD (Advanced front-end) and SILK (developed by Skype).

The VAD of the G.729-B standard generates a binary VAD decision for each frame as a function of four relevant parameters extracted from the audio signal. In a simplified way, these parameters are directly related to energy and spectral components of a voiced frame. The G.729-B is used in the majority of audio transmission applications, becoming the most popular VAD technique. Therefore, the G-729-B is today the best standard for comparative studies in most scientific articles dealing with VAD.

In the 1st contribution, we propose a VAD scheme based on adaptive threshold while maintaining the False Acceptance Rate at a nominal value. As well known in the binary decision theory, the error rate, denoted "False Acceptance Rate", is related to the probability of misclassified a frame of silence as Active Voice. The basic idea is to perform sequential tests, based on full band energy, in order to reject or to accept the frame under investigation as active voice region. The most interesting feature of the proposed algorithm concerns its ability to dynamically update the noise level estimator, according to the current environment. Taking into account the long-term stationary property of the speech, we also developed a smoothing procedure to discard discontinuities that may appear in the processed signal.

The performance of the proposed approach has been evaluated and compared to the VAD of the G.729-B in several situations including various environmental acoustic noises with different SNRs. Analysis of the results has been performed using the NOIZEUS experimental database as well as real recorded signals.

The 2nd contribution consists of implementing the proposed approach on a microcontroller-based system, in order to:

- *Ensure the robustness of the algorithm,*
- *Evaluate its implementation complexity*
- *Validate the real time operation mode*

In this context, various tests were conducted in real time mode via the development tools available on the microcontroller system (STM32F7). These tools allowed performing real-time monitoring of several signal parameters in realistic situations. By this way, we were able to accurately determine the processing time (Latency) required to generate a final decision for each frame. The real-time analysis allowed us to obtain a global latency of 4 µs, which seems sufficient to guarantee real-time operation regarding to the common sampling frequencies of speech processing systems (8 kHz to 16 kHz).

Table des matières

Liste des Tableaux

.

Liste des Figures

Liste des Symboles

FA : False Acceptance

FAex : False Acceptance experimental

dB : décibel

G(.) : fonction de Gauss

Γ(.) : fonction Gamma

exp(.) : fonction Exponentiel

N_0 : Nombre des trames d'initialisation

a: Coefficient d'ajustement

ΔE : l'écart d'énergie

H_0 : Hypothèse en absence de voix

H_1 : Hypothèse en présence de la voix

Log : Logarithme

Res[] : le résidu

Φ : la fonction génératrice des moments

R_x : l'autocorrélation

Liste des Acronymes

ADC:	Analog-to-Digital Converter
AFE:	Advanced Front End
AHB:	Advanced High-performance Bus
AMR:	Adaptive Multi-Rate
APB:	Advanced Peripheral Bus
ARM:	Advanced RISC Machines
ARP:	Auditory Research Platform
AWGN:	Additive White Gaussian Noise
CAN:	Controller area network
CEC :	Consumer Electronics Control
CNG:	Comfort Noise Generation
CNN:	Convolutional Neural Network
CPU:	Central Processing Unit
DAC:	Digital-to-Analog Converters
DBN:	Deep-Belief Network
DC:	Direct current
DCMI:	Digital Camera Interface
DD:	Décision Dirigée
DFT:	Discrete Fourier Transform
DMA:	Direct Memory Access
DNN:	Deep Neural Network
DSP:	Digital Signal Processor
EDI:	Environnement de Développement Intégré
ETM:	Automatic Teller Machine
ETSI:	European Telecommunications Standards Institute
FA:	False Acceptance
FAex:	False Acceptance Experimental
FEC:	Front End Clipping
FIFO:	First In First Out
FLL:	Frequency-Locked Loop
FPGA:	Field Programmable Gate Array
GPIO:	General Purpose Input Output
HAL:	Hardware Abstraction Layer
HDMI:	High-Definition Multimedia Interface
I2C:	Inter Integrated Circuit Bus
I2S:	Inter Integrated Sound
IAR:	Ingenjörsfirman Anders Rundgren
IDE:	Integrated Development Environment
IEEE:	Institute of Electrical and Electronics Engineers
IID:	Independent and Identically Distributed

IEEE:	Institute of Electrical and Electronics Engineers
IID:	Independent and Identically Distributed
LCD:	Liquid Crystal Display
LRT:	Likelihood Ratio Test
LSB:	Least Significant Bit
LUT:	Look Up Tables
MCLK:	Master clock
MCU:	Microcontroller Unit
MEMS:	Micro Electro Mechanical System
MGF:	Moment Generating Function
MIC:	Modulation par impulsions et codage
MIPS:	Million of Instructions Per Second
ML:	Maximum Likelihood
MPU:	Memory Protection Unit
MSB:	Most Significant Bit
MSC:	Mid Speech Clipping
NDS:	Noise Detected as Speech
OS:	Order Statistics
PCM:	Pulse Code Modulation
pdf:	Probability Density Function
RAM:	Random Access Memory
RF:	Radio Frequency
RG45:	Registered Jack 45
RGB:	Red Green Blue
RNN:	Recurrent Neural Network
RTC:	Real Time Clock
RTOS:	Real Time Operating System
SAI:	Serial Audio Interface
SDCT:	Sequential Detection of Change Test
SDMMC:	Secure Digital and Multi Media Card
SMG:	Special Mobile Group
SNR:	Signal to Noise Ratio
SPDIF:	Sony/Philips Digital Interface Format
SPI:	Serial Peripheral Interface
SRAM:	Static Random Access Memory
STM:	STMicroelectronics
SWD:	Serial Wire Debug
SYSCLK:	System Clock
TDM:	Time Division Multiplexin
TDT:	Tucker Davis Technologies

TFT:	Thin Film Transistor
TIMx:	Timers
TPA:	Trace Port Analyzer
TR:	True Rejection
UART:	Universal Asynchronous Receiver Transmitters
USART:	Universal Synchronous/Asynchronous Receiver Transmitters
USB:	Universal Serial Bus
VAD:	Voice Activity Detection
VoIP:	Voice over Internet Protocol
XSG:	Xilinx System Generator

Chapitre 1:
Introduction aux Concepts Parole

Résumé

Dans ce chapitre nous introduisons les concepts de base de la parole et le traitement numérique des signaux vocaux, nous définissons le principe de fonctionnement d'un module VAD, son importance et ses divers domaines d'applications, ensuite, les problèmes rencontrés relativement aux bruit de fond stationnaire ou non-stationnaire, à la fin, nous expliquons nos deux contributions ; la technique VAD développée et son implémentation dans un microcontrôleur.

1.1 Généralités

Par définition, la détection d'activité vocale consiste à déterminer les segments sonores avec et sans parole dans une conversation ou dans une simple phrase. Cette étape est cruciale dans la plupart des systèmes de traitement de la parole car elle permet d'identifier et de délimiter avec précision l'information utile, à savoir la voix. En effet, les données superflues, c'est-à-dire dépourvues de parole, n'ont pas besoin d'être traitées ni transmises. Dans certaines applications, il est même primordial qu'elles ne subissent pas le traitement réservé à la parole, comme par exemple lors de l'amplification de la voix. Lorsqu'il y a peu ou pas de bruit, cette opération est aisée. Dans le cas d'un bruit important, c'est-à-dire quand le rapport signal à bruit est faible, des parties de parole sont complètement ensevelies sous ce dernier et il devient alors difficile de détecter correctement 1'activité vocale. Ceci peut être observé sur la Figure 1.1.

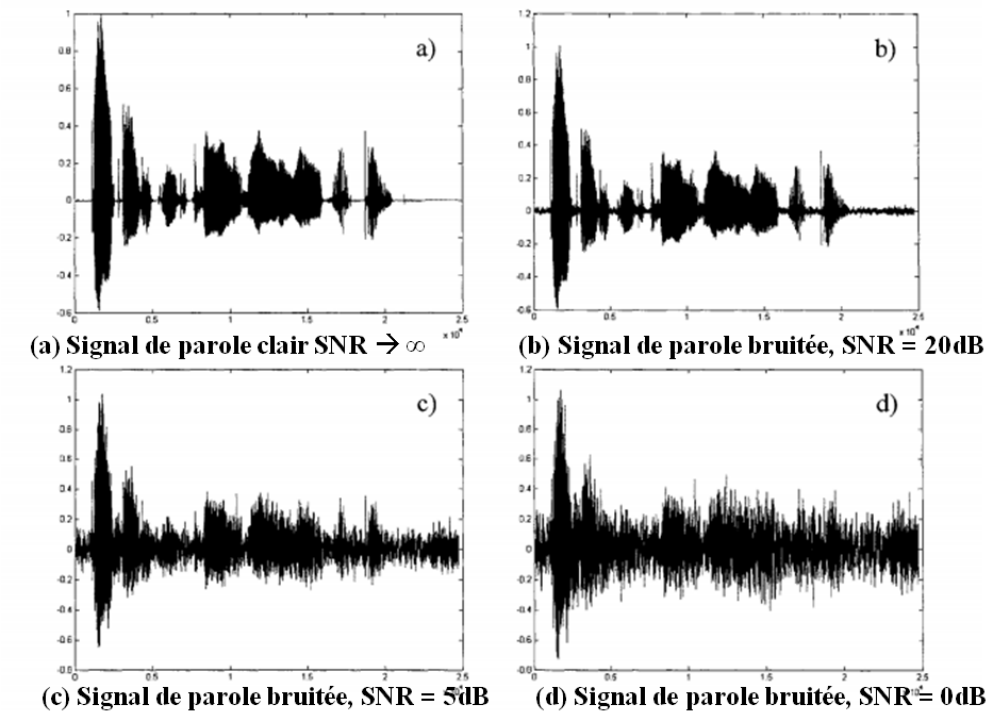


Figure 1.1: Exemple de signaux de parole bruitée

On rappelle que le SNR est défini par

$$SNR = 10log_{10}\left(\frac{P_x}{P_b}\right)$$

Avec:

Px : la puissance du signal initial

Pb : la puissance du signal de bruit

Ainsi plus le SNR est faible, plus le signal initial est noyé dans le bruit.

Le système réalisant la délimitation de la parole s'appelle un détecteur d'activité vocale. Son principe est décrit par la Figure 1.2.

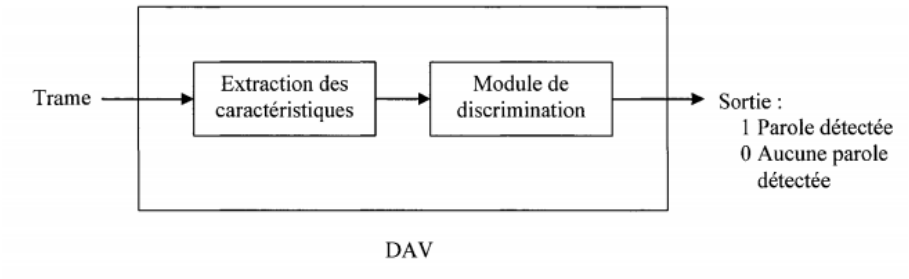


Figure 1.2: Schéma fonctionnel d'un VAD

Le signal à étudier est tout d'abord découpé en trames de longueur fixe. Chaque trame arrive ensuite à l'entrée du VAD. Ce dernier en extrait certaines caractéristiques. À l'aide de règles de décision portant sur les valeurs de ces paramètres, il prend alors une décision quant à l'état de la trame. La sortie du VAD ainsi obtenue est une variable logique. Si elle est unitaire, la trame contient de la parole. On dit alors que cette trame est active ou PAROLE. Si la sortie du VAD est nulle, on dit que la trame est inactive, ou BRUIT, ou encore, par abus de langage, qu'il s'agit d'une trame de SILENCE. La Figure 1.3 montre la sortie du VAD dans le cas du signal de parole claire:

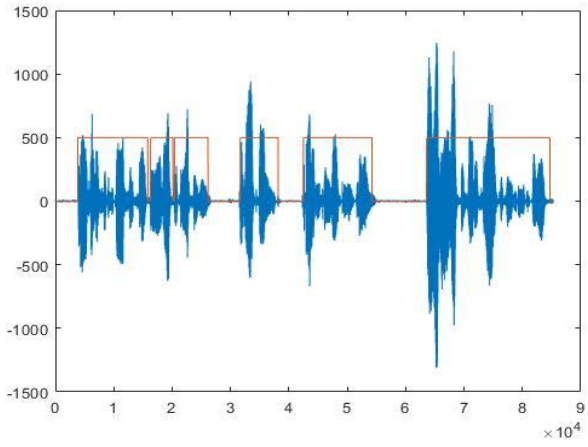


Figure 1.3: Sortie du VAD pour un signal de parole clair

Les VAD sont surtout utilisés dans les télécommunications, au sein des codeurs de parole pour la téléphonie fixe, mobile, multimédia ou voix sur IP. Ils permettent d'éviter le traitement de l'information inutile, bruit de fond ou silence, et ainsi de libérer le canal de transmission pour d'autres applications ou d'autres communications. En général, lorsqu'une trame est déclarée inactive par le VAD, le codeur de parole ne la transmet pas mais envoie à la place un «descripteur d'insertion de silence». Quand le décodeur, de l'autre côté du canal de transmission, reçoit ce type d'information, il déclenche alors le« générateur de bruit de confort», simulant le bruit de fond enregistré par le codeur. Il donne ainsi l'impression aux interlocuteurs que leur conversation entière a été transmise. Le système regroupant le VAD, le «descripteur d'insertion de silence» et le« générateur de bruit de confort» permet donc à la fois une conversation agréable et une économie des transmissions, d'où l'importance d'un algorithme de détection d'activité vocale efficace. Il est à noter qu'en cas de doute le VAD devrait toujours indiquer la présence de parole afin de conserver l'information utile et ainsi la bonne qualité du message. La plupart des méthodes proposées dans la littérature ont été développées pour les télécommunications.

1.2 Notions de base de la parole

1.2.1 La parole

- **Définition d'un son**

Un son est un phénomène physique qui fait réagir notre cerveau, c'est une sensation auditive provoquée par une onde acoustique. D'un point de vue physique, il s'agit d'une vibration qui se propage dans un milieu matériel solide, liquide ou gazeux.

La perturbation associée à une onde sonore concerne la pression interne d'un milieu matériel. Ainsi plus la pression acoustique est importante, plus le volume sonore est grand. Depuis sa source, l'onde mécanique modifie la valeur de cette pression en chaque point de son trajet. Grâce à l'excitation mécanique, les molécules ayant reçu une impulsion se mettent en mouvement et entrent en collision avec les molécules voisines auxquelles elles communiquent le même mouvement. Une zone de compression est alors créée. À cause du choc, les premières reculent et dépassent leur position de repos, c'est pourquoi une détente succède toujours à une compression, tandis qu'une autre zone décompression se forme plus loin. Il s'établit ainsi des oscillations. Le mouvement des molécules voisines étant limité pour les mêmes raisons, elles oscillent à leur tour. Petit à petit, ce mouvement se propage, créant ainsi

une onde sonore à l'origine du son. L'onde sonore créée par le mouvement oscillatoire des particules se disperse autour de la source émettrice selon une sphère. Plus l'onde sonore ne s'éloigne de la source, plus la surface de la sphère augmente et plus l'intensité diminue. La transmission s'accompagne d'une dissipation d'énergie sous forme de chaleur, ce qui provoque l'amortissement de l'onde avec la distance. La propagation du son se fait à une vitesse dépendant des caractéristiques et des conditions de température et de pression du milieu [1]. On peut diviser les sons en deux catégories : les sons purs : ils correspondent à des mouvements d'oscillations des particules pures, c'est à dire une sinusoïdale parfaite. Un son pur est en fait constitué d'une fréquence unique. Il est très peu répandu dans la nature. Les sons complexes : ils peuvent contenir des éléments périodiques, transitoires ou aléatoires. Les éléments sonores périodiques (cas de la parties ou tenue d'une note d'un instrument de musique) sont caractérisés par leur fréquence de base, dite fondamentale, et leurs harmoniques, multiples de la fréquence fondamentale. Les harmoniques déterminent le timbre d'un son et selon leur nombre et leur fréquence, on peut ainsi distinguer le violon de la flûte. Les éléments transitoires (cas de l'attaque d'une note d'un instrument de musique) sont plus difficilement caractérisables. Les éléments aléatoires (cas du bruit de la turbulence générée par un écoulement) sont souvent appelés bruit en traitement des signaux et sont caractérisés par leur contenu fréquentiel et leur densité spectrale de puissance. Un cas extrême est le bruit « blanc » dont le spectre contient toutes les fréquences avec la même densité spectrale de puissance [1].

- **Définition de la parole**

La Figure 1.4 présente les principaux organes utilisés lors de la production de la parole :

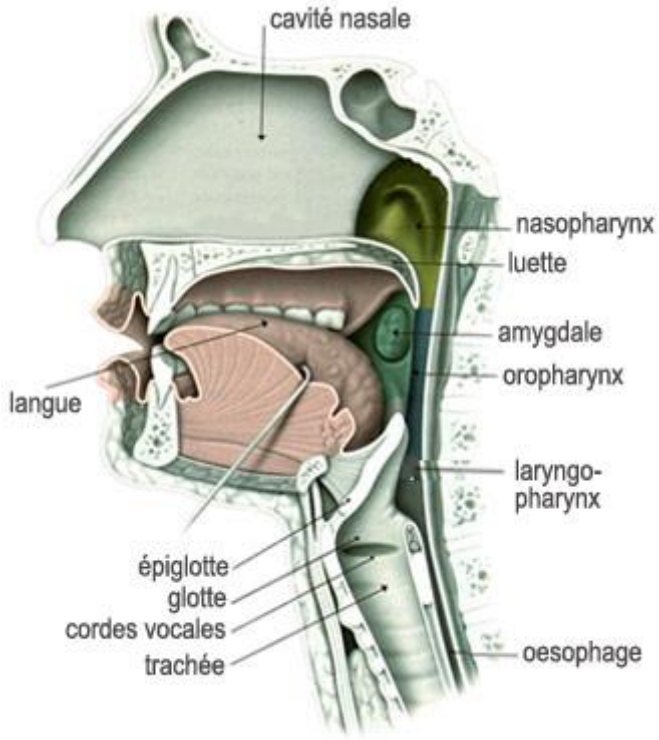


Figure 1.4: Les principaux organes de la production de la parole

L'air contenu dans les poumons traverse la trachée et arrive dans le larynx où il rencontre les cordes vocales. Ce contact entraîne un mouvement des cordes vocales (phonation) qui lui-même provoque l'ouverture et la fermeture rapides de la glotte. Cette vibration engendre alors une onde sonore, prémisse du signal de parole. Le pharynx ainsi que les cavités buccales et nasales modifient ensuite les caractéristiques de cette onde, en atténuant ou en amplifiant certaines fréquences. Ils ont donc un rôle de résonateur. Le son laryngé devient finalement de la parole lorsqu'il a été modulé par la position de la langue, du voile du palais, des dents et des lèvres, [2].

La parole fait partie des sons complexes. Ainsi, chaque trace vocale est caractérisée par sa fréquence fondamentale, appelée aussi pitch, et par ses harmoniques. Le pitch est directement lié au nombre de fois que la glotte s'ouvre et se ferme par seconde. Pour certains sons, les cordes vocales ne vibrent pas, il n'y a alors pas de pitch. Les harmoniques, quant à eux, dépendent de l'endroit où a eu lieu la résonance. On appelle formants les zones fréquentielles entourant ces fréquences de résonance. Les traces vocales se distinguent aussi par leur forme (durée, intensité...) et sont particulières à chaque individu, idée fondamentale de la reconnaissance du locuteur [3]. La parole étant un son, elle se propage et arrive finalement à l'oreille. L'être humain entend alors ce son s'il est compris entre 20Hz et20 kHz.

La Figure 1.5 schématise l'oreille humaine [4]:

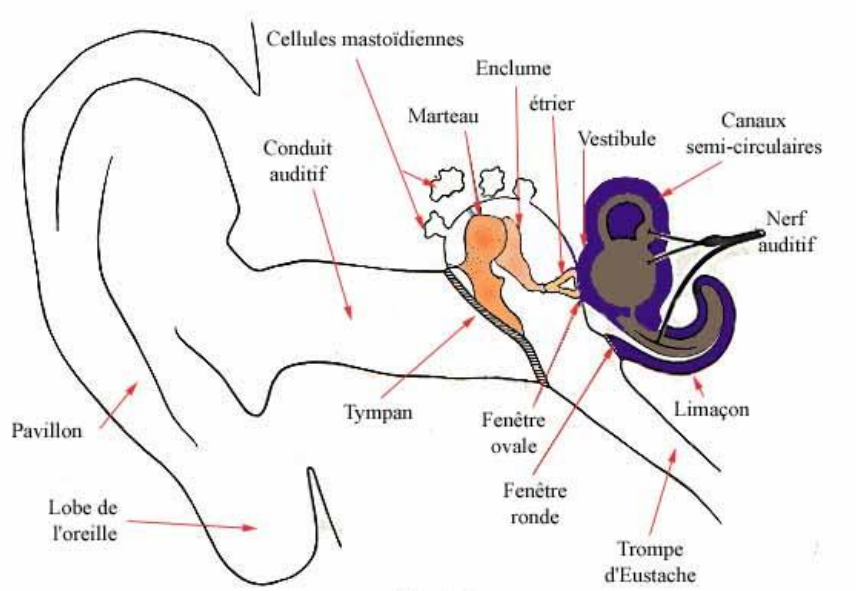


Figure 1.5: Oreille humaine.

Lorsqu'un son vient heurter les tympans, ils se mettent à onduler puis la vibration est transmise jusqu'à l'oreille interne dans laquelle se trouve le nerf auditif. Ce dernier transmet l'information au cerveau et une sensation sonore est alors ressentie [1].

- **Différents types de parole**

D'après Tetschner [5], les sons parlés peuvent être regroupés en deux catégories : les sons voisés : on dit qu'un son est voisé lorsque les cordes vocales vibrent de façon quasi-périodique, c'est-à-dire lorsqu'il y a phonation. Le signal de parole est alors caractérisé par son pitch. Les sons non-voisés : on dit qu'un son est non-voisé s1 le phénomène de phonation est absent. Il ne possède donc pas de pitch. Une autre manière de classer les sons parlés est d'utiliser leurs caractéristiques articulatoires. Il existe deux unités phonétiques [6] : les voyelles : elles sont produites en laissant passer l'air librement dans le conduit vocal et ceci sans obstruction d'aucune sorte. Elles provoquent la vibration des cordes vocales et font donc partie des sons voisés. Les consonnes : elles sont produites par une obstruction partielle ou totale du conduit, par exemple à l'aide du palais, de la langue, des lèvres... Elles peuvent être voisées ou non. Il existe une troisième façon de diviser les sons parlés : le modèle phonologique mais nous ne l'aborderons pas ici car cette classification est rarement utilisée dans le domaine de la détection d'activité vocale [6].

1.2.2 Le traitement numérique des signaux

Depuis plusieurs décennies, la technologie des microprocesseurs s'est développée rapidement. Le traitement numérique des signaux est alors devenu très populaire. En effet, comparé à un système analogique, un système numérique est plus rapide, plus flexible, généralement plus simple à concevoir et surtout plus économique. La majorité des traitements sur les signaux s'effectue aujourd'hui de manière numérique ; c'est précisément le cas du détecteur d'activité vocale (VAD). Les signaux de parole, et les sons en général, sont des signaux analogiques temporels. Afin de les manipuler numériquement, il faut d'abord les convertir en signaux numériques. Une fois qu'ils ont été traités, il est nécessaire de les retransformer en signaux analogiques afin que le signal de sortie soit de nouveau un son. Le schéma fonctionnel d'un tel système est décrit par la Figure 1.6 [7]:

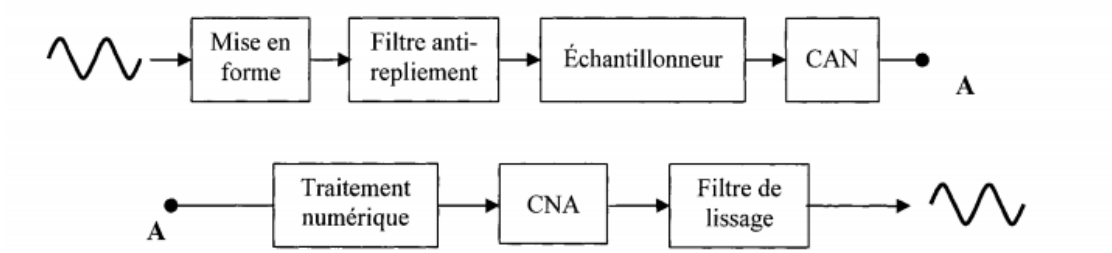


Figure 1.6: Schéma fonctionnel d'une chaine de traitement numérique des signaux

D'après Oppenheim et Schafer [7], les différentes étapes mises en œuvre dans un système numérique, présenté par la Figure 1.6, peuvent être décrites de la manière suivante:

Mise en forme : dans la plupart des cas, la donnée à traiter est une grandeur physique. Un capteur approprié est alors utilisé pour transformer cette grandeur physique en tension ou en courant. Le signal est ensuite mis en forme par des circuits spécialisés de conditionnement. Échantillonneur: l'échantillonnage d'un signal consiste à prélever des échantillons sur le flux continu d'information analogique à des intervalles de temps discrets et constants Y: . Le choix de cette période est important pour assurer la précision du signal restitué après traitement. Il apparaît clairement que plus les échantillons sont rapprochés, plus la précision du signal sera importante, par contre un grand nombre d'échantillons par seconde conduira à un flux de données plus difficile à traiter. Le choix de la fréquence d'échantillonnage fe doit respecter le théorème de Nyquist afin d'assurer une bonne restitution du signal et d'éviter le repliement spectral.

Pour éviter toute perte d'information et donc pour que le signal échantillonné puisse être reconstruit à partir de ses échantillons, il faut et il suffit que la fréquence d'échantillonnage fe soit plus grande ou égale à deux fois la fréquence max du signal à échantillonner :

fe ≥ 2 fréquence max = fréquence de Nyquist

Filtre anti-repliement : même en respectant le théorème de Nyquist, on peut être en présence de repliement spectral. En effet, le signal à échantillonner est le plus souvent entaché de bruit, dont le spectre est généralement plus étendu que celui du signal utile. L'échantillonnage a alors pour effet de superposer, au spectre basse fréquence du signal utile, certaines parties du spectre du bruit. Les composantes du bruit au dessus de la fréquence d'échantillonnage sont «repliées» sur le spectre du signal utile. Pour enlever cet effet perturbateur, il faut éliminer la partie haute fréquence du bruit. Pour cela, on utilise un filtre passe-bas de fréquence de coupure égale, ou parfois inférieure, à la moitié de la fréquence d'échantillonnage. Ce filtre est appelé filtre anti-repliement et doit précéder l'échantillonneur.

Convertisseur Analogique-Numérique (CAN) : un signal numérique est un signal discret quantifié, c'est-à-dire que son amplitude est, elle aussi, discrétisée. C'est cette quantification qui rend possible la représentation d'un signal par un nombre fini de bits. Le CAN permet donc de transformer le signal discret en un signal numérique. La conversion n'étant pas instantanée, il est nécessaire d'insérer entre la sortie de l'échantillonneur et l'entrée du CAN un bloqueur d'ordre zéro qui a pour fonction de maintenir suffisamment longtemps la valeur d'un échantillon. Le temps de blocage doit bien sûr être supérieur au temps de conversion pour que le CAN puisse convertir les données correctement, le temps de conversion étant le temps qui s'écoule entre l'impulsion de demande de conversion et la stabilisation des données dans le tampon.

Traitement numérique : en pratique, les opérations sur les signaux numériques sont effectuées à l'aide d'un Digital Signal Processor (DSP). Le DSP manipule les données obtenues à la sortie du CAN afin de réaliser la fonction qu'il doit assurer. À sa sortie, les données modifiées sont du même type qu'à son entrée, c'est-à-dire numériques. Il est caractérisé par plusieurs paramètres comme son architecture, sa mémoire, son jeu d'instructions, ses registres d'adressage, sa fréquence d'horloge... C'est à ce niveau que le détecteur d'activité vocale intervient. En effet, une fois celui-ci mis au point, il suffit de coder son algorithme en assembleur sur le DSP.

Ce projet de recherche s'agit de développer un détecteur d'activité vocale efficace basé sur des décisions statistiques, qui assure un taux de fausse acceptation fixe en absence de la voix, ensuite, notre approche a été comparée avec le VAD standard du G.729 dans un environnement stationnaire et non-stationnaire, à la fin, l'algorithme a été implémenté dans un microcontrôleur pour assurer le fonctionnement en temps réel.

Convertisseur Numérique-Analogique (CNA) : une fois le traitement effectué, il est nécessaire d'avoir recours à une conversion numérique-analogique afin que le signal de sortie de la chaîne soit encore un son. À l'entrée du CNA, il y a des échantillons dont l'amplitude est binaire. Ce convertisseur va permettre de convertir cette amplitude binaire en une amplitude analogique. Sa sortie est donc constituée d'une suite d'impulsions rectangulaires juxtaposées et de largeur égale à la période d'échantillonnage.

Filtre de lissage : le signal de sortie du CNA n'est pas encore du même type que le signal d'entrée de la chaîne de traitement. Pour enlever cet effet rectangulaire, il faut lisser ce signal. Pour cela, on utilise un filtre passe-bas de fréquence de coupure égale à la moitié de la fréquence d'échantillonnage. Ce filtre est appelé filtre de lissage et doit suivre directement le CNA. Ainsi le signal, après traitement numérique, est du même type qu'au départ, par contre ses caractéristiques ont été modifiées comme désiré.

2.3 Les domaines d'application du VAD

Les VAD sont utilisés dans de nombreux domaines du traitement de la parole. Les méthodes VAD ont été décrites dans la littérature pour plusieurs applications dont les services de communication mobile [8], transmission de la parole en temps réel sur Internet [9] ou réduction du bruit pour les aides auditives numériques [10]. A titre d'exemple, un VAD réalise la compression du silence dans les systèmes de télécommunications mobiles modernes en réduisant le débit binaire moyen en utilisant le mode de transmission discontinue (DTX). Beaucoup d'applications pratiques, telles que la téléphonie mobile, utilisent détection de silence et injection de bruit de confort pour une efficacité de codage plus élevée. On montre ci-dessous une brève description des applications VAD les plus importantes dans le traitement de la parole.

1.2.1 Codage de la parole

Le VAD est largement utilisé dans le domaine de la communication vocale pour obtenir une efficacité de codage de la parole élevée et une transmission à faible débit binaire. Les concepts de détection de silence et de génération de bruit de confort conduisent à des techniques de codage de la parole avec deux différents modes de fonctionnement : i) le codec vocal actif, et ii) les modes de suppression de silence et de génération de bruit de confort.

L'Union internationale des télécommunications (UIT) a adopté un algorithme de codage de la parole de qualité à péage appelé G.729 [11] pour fonctionner en combinaison avec un module VAD en mode discontinu (DTX). La Figure 1.7 montre un schéma de principe d'un codec vocal.

Le codeur de parole à plein débit est opérationnel pendant la parole vocale active, mais un schéma de codage différent est utilisé pour le signal vocal inactif, utilisant moins de bits et résultant en un taux de compression moyen global plus élevé.

Un autre standard pour le mode discontinu DTX est le codeur de parole AMR ETSI (Adaptive Multi-Rate) (European Telecommunications Standards Institute) [12] développé par le Special Mobile Group (SMG) pour le système GSM. La norme spécifie deux options pour le VAD à utiliser dans le système de télécommunications cellulaires numériques, la transmission vocale bimode atteint un débit binaire significatif du codage numérique de la parole car environ 60% du temps, le signal transmis contient juste du silence dans une communication téléphonique.

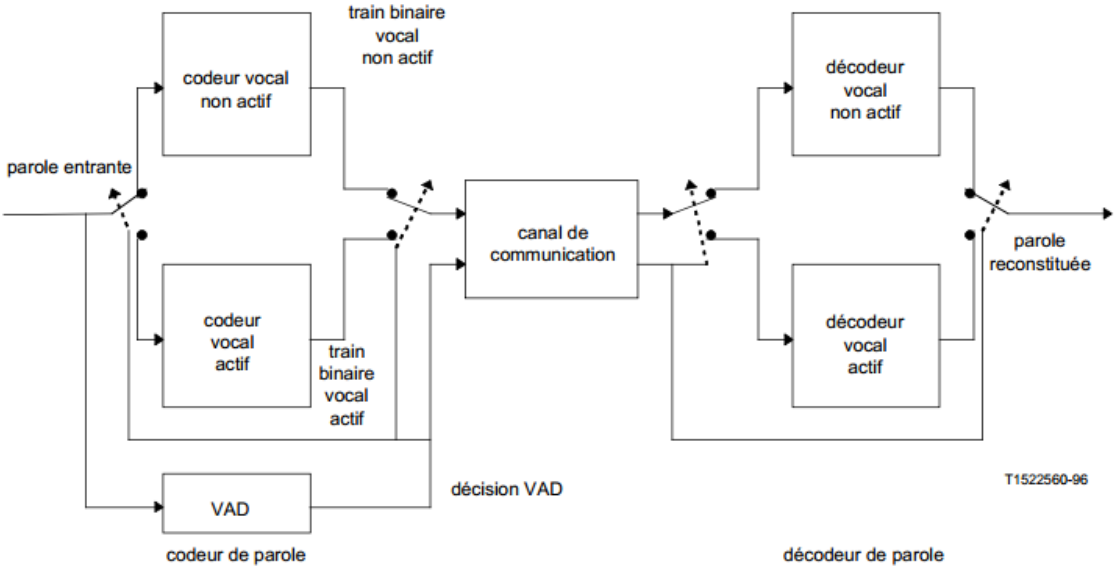


Figure 1.7: Schéma de communication de parole VAD

1.2.2 Rehaussement de la parole

Le rehaussement de la parole vise à améliorer les performances des systèmes de communication vocale dans des environnements bruyants. Il traite principalement la suppression du bruit de fond d'un signal bruyant. Une difficulté dans la conception des systèmes efficaces d'amélioration de la parole est le manque des modèles statistiques explicites pour le signal de parole et le processus de bruit, de plus, le signal de la parole, et éventuellement aussi le processus de bruit, ne sont pas des processus strictement stationnaires.

L'amélioration de la parole suppose normalement que la source de bruit est additive et non corrélée avec le signal de parole propre.

1.2.3 Reconnaissance vocale

La VAD est une technique très utile pour améliorer les performances des systèmes de reconnaissance. Un module VAD est utilisé dans la plupart des systèmes de reconnaissance dans le cadre du processus d'extraction des caractéristiques pour l'amélioration de la parole. Les statistiques du bruit telles que son spectre sont estimées pendant les périodes non vocales afin d'appliquer les algorithmes d'amélioration de la parole (soustraction spectrale ou filtre de Wiener). D'autre part, La suppression de la trame non vocale (Frame Dropping FD) est également une technique fréquemment utilisée pour réduire le nombre d'erreurs d'insertion causées par le bruit. Il consiste à supprimer les périodes de non-parole (sur la base de la décision VAD) de l'entrée du dispositif de reconnaissance vocale, cela réduit le nombre d'erreurs d'insertion en raison du bruit qui peut être une source d'erreur grave dans des conditions entraînement / test très inadéquates. La Figure 1.8 montre un exemple d'un système de reconnaissance vocale robuste intégrant la réduction du bruit spectral en utilisant la suppression de la trame non-parole.

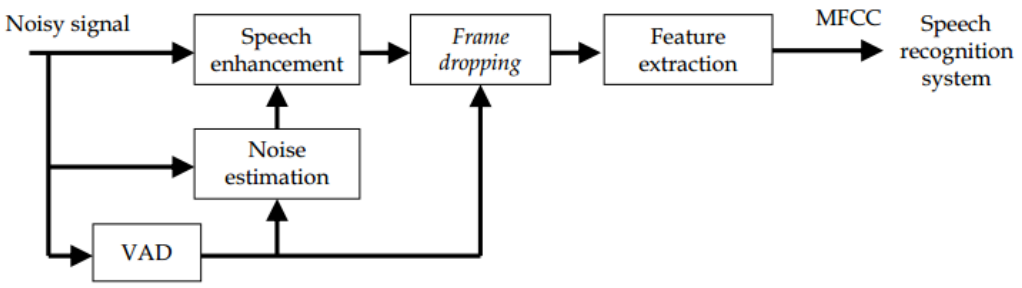


Figure 1.8: Extraction de caractéristiques avec réduction du bruit spectral

1.4 Contexte et problématique

Un problème important dans de nombreux domaines du traitement de la parole est la détermination de la présence de périodes de parole dans un signal donné. Cette tâche peut être identifiée comme un problème d'hypothèse statistique et son objectif est de déterminer à quelle catégorie ou classe appartient un signal donné. La tâche de classification n'est souvent pas aussi triviale qu'il n'y paraît car le niveau croissant de bruit de fond dégrade l'efficacité du classificateur, conduisant ainsi à de nombreuses erreurs de détection. La sélection d'un vecteur de caractéristiques adéquat pour la détection du signal et d'une règle de décision robuste est un problème difficile qui affecte les performances des VAD fonctionnant dans des conditions de bruit. La plupart des algorithmes sont efficaces dans de nombreuses applications mais provoquent souvent des erreurs de détection principalement dues à la perte du pouvoir discriminant de la règle de décision à de faibles niveaux de SNR [11] [12]. Par exemple, un simple détecteur de niveau d'énergie peut fonctionner de manière satisfaisante dans des conditions de rapport signal / bruit (SNR) élevé, mais échouerait de manière significative lorsque le SNR chute. La VAD est plus critique dans les environnements de bruit non stationnaires car il est nécessaire de mettre à jour les statistiques de bruit constamment variables affectant fortement une erreur de classification aux performances du système.

1.5 Contributions

Dans cette thèse, nous nous sommes intéressés à la mise au point d'un Détecteur d'Activité Vocale (VAD) robuste, efficace et peu complexe. Par ailleurs, nous avons abordé la problématique liée à l'aspect temps réel, en proposant une solution qui répond aux impératifs d'implémentation.

Dans la 1ère contribution, nous avons proposé une technique VAD basée sur le principe du seuillage adaptatif, tout en maintenant le taux de fausse acceptation (False Acceptance Rate) à une valeur nominale. Comme clairement établi dans la théorie de la décision binaire, le taux d'erreur, noté « False Acceptance Rate », s'apparente à la probabilité de déclarer une trame de silence comme étant Voix Active. S'agissant des modèles de distribution du bruit de fond et du signal composite (Voix plus Bruit), nous avons considéré, respectivement, le modèle

AWGN (Additive White Gaussienne Noise) et la distribution de Laplace ; ces deux fonctions de densité de probabilité (pdf) étant les plus couramment utilisées dans ce domaine.

Le fondement de la méthode proposée s'appuie sur la mise en œuvre de tests statistiques séquentiels basés sur l'énergie en pleine bande, afin de rejeter ou d'accepter la trame en cours d'investigation en tant que région vocale active. La caractéristique principale de l'algorithme réside dans sa capacité à mettre à jour, dynamiquement, l'estimateur du niveau de bruit en fonction de l'environnement en cours. L'estimation du niveau de puissance locale se calcule en tenant compte des trames les plus récemment déclarées « Silence ». En considérant l'hypothèse de stationnarité du signal parole, nous avons également développé une procédure de lissage (Smoothing) afin de minimiser les discontinuités pouvant avoir lieu dans les régions vocales et non vocales.

Les performances de l'approche proposée ont été évaluées et comparées au VAD de la norme G.729-B dans plusieurs situations incluant divers bruits acoustiques environnementaux avec différents SNR. L'analyse des résultats a été réalisée aussi bien en situation stationnaire qu'en présence d'environnements non stationnaires. Toutes les expérimentations ont été conduites en utilisant la base de données expérimentale NOIZEUS ainsi que des signaux vocaux enregistrés dans des environnements réalistes.

La 2ème contribution consiste à implémenter l'approche proposée sur un système à base de microcontrôleur, afin de :

- S'assurer de la robustesse de l'algorithme,
- Evaluer sa complexité d'implémentation
- S'assurer de son fonctionnement en temps réel

Dans ce contexte, divers tests on été conduits en temps réel à l'aide d'outils d'analyse et de développement accompagnant le système à microcontrôleur (STM32F7). Ces outils nous ont permis d'effectuer un monitoring en temps réel dans des diverses situations, en visualisant le signal en cours d'enregistrement, le seuil adaptatif par rapport à l'énergie en pleine bande ainsi que la décision VAD pour chaque trame traitée. Grace à cette façon de procéder, nous avons pu déterminer avec précision le temps de traitement (Latence) nécessaire à la génération de la décision finale pour

chaque trame. L'analyse en temps réel du système VAD a permis d'obtenir une latence de 4 µs, ce qui semble amplement suffisant pour garantir un fonctionnement en temps réel en tenant compte des fréquences d'échantillonnage les plus couramment utilisées en traitement de la parole (8 kHz à 16 kHz).

1.6 Plan de lecture du manuscrit

Le plan du manuscrit est organisé comme suit :

Dans le chapitre 1, nous avons abordé des concepts généraux concernant la parole et le phénomène du son, ainsi que le traitement numérique du signal vocal basé principalement sur des convertisseurs (A/N) et (N/A). Ensuite, nous avons cité quelques domaines d'application de la détection d'activité vocal (VAD), comme ; le codage de la parole, le rehaussement de la parole et la reconnaissance vocale. A la fin, nous avons introduit en détail nos deux contributions majeures qui feront l'objet des chapitres 3 et 4 respectivement.

Dans le chapitre 2, nous avons fait un exposé exhaustif des VAD standards les plus utilisés dans le monde industriel et les plus référencées dans la recherche scientifique, à savoir ; le VAD de la norme G.729, le VAD AMR (Adaptive multi-rate), le VAD AFE (Advanced front-end) ainsi que celui développée par Skype (SILK).

Dans le chapitre 3, nous avons exposé l'algorithme VAD proposé tant sur le plan de la structure algorithmique et du schéma de calcul que sur le plan du développement mathématique qui a conduit à l'obtention d'une expression compacte pour le taux de fausse acceptation (*Fa*).

Dans le chapitre 4, nous avons abordé les aspects liés à l'implémentation de notre technique VAD via un système à base de microcontrôleur STM32F7. Dans cette section, nous avons donné tous les détails relatifs à la solution implémentée ainsi qu'aux tests exécutés en mode « temps réels » pour la validation de l'approche proposée.

Chapitre 2:
Méthodes VAD industrielles

Résumé

Dans ce chapitre nous exposons les techniques VAD standards les plus implémentées dans le monde industriel, mais également les plus référencées en tant que base comparative au niveau de la littérature scientifique. Nous commençons par le standard VAD G.729-B approuvé par l'UIT-T en 1996, ensuite le VAD AMR (Adaptatif Multi Rate) et le VAD AFE (Advanced front-end). A la fin, nous donnerons un aperçu succinct sur le VAD SILK, développé par Skype.

2.1 Introduction

2.2 Les méthodes VAD industrielles

2.2.1 UIT G.729 VAD [11]

Le diagramme fonctionnel de l'algorithme VAD est donné à la Figure 2.1. L'algorithme VAD fonctionne sur des trames de parole numérisée. Les trames sont traitées dans l'ordre chronologique et sont numérotées en séquence dès le début de chaque conversation/enregistrement.

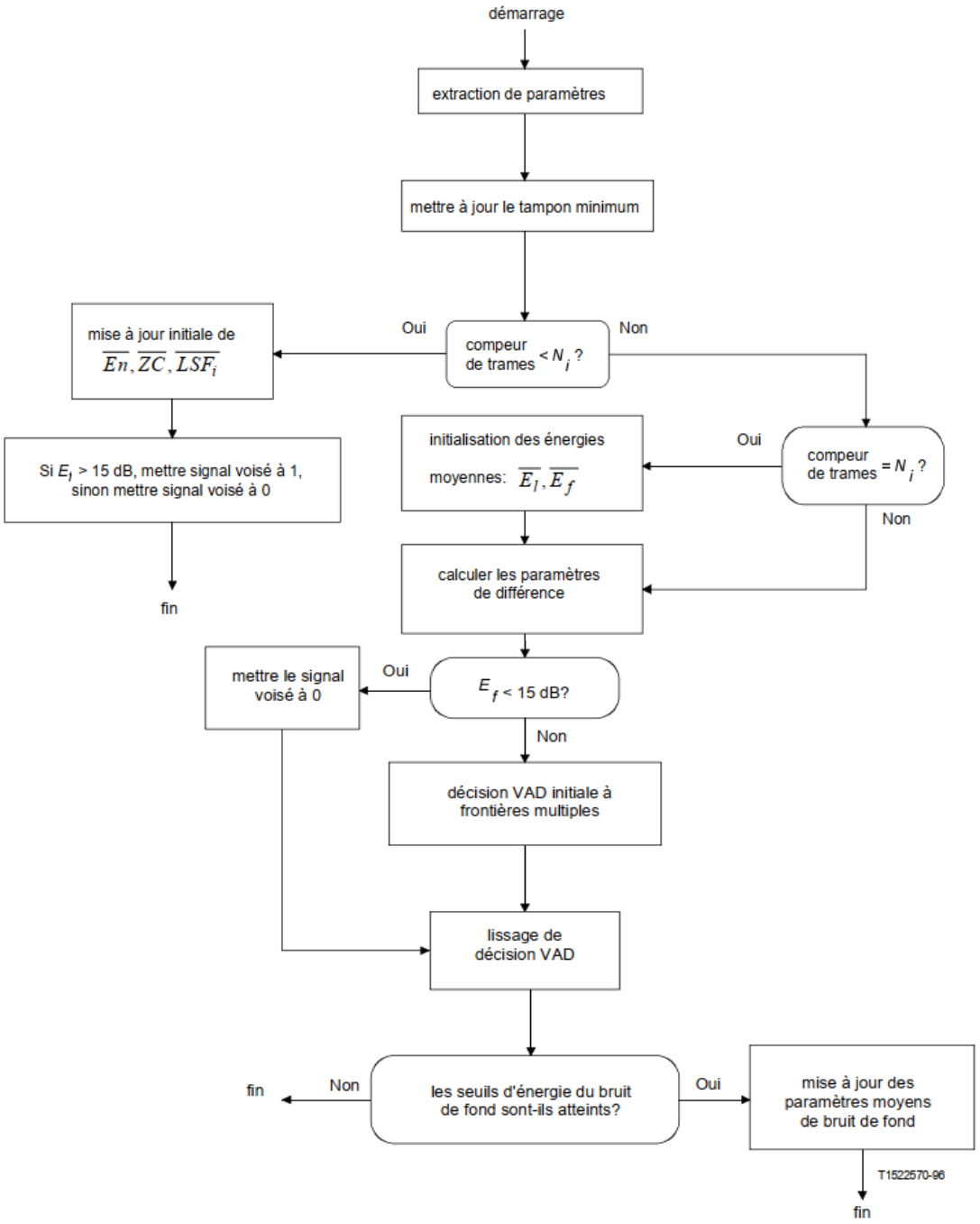


Figure 2.1: Schéma fonctionnel du VAD G.729

A la première étape, quatre options paramétriques sont extraites du signal d'entrée.
L'extraction des paramètres est partagée par le module du codeur vocal actif et le codeur vocal non actif pour améliorer l'efficacité des calculs informatiques. Les paramètres sont les énergies dans la pleine bande de fréquences et dans la bande de fréquences basses, l'ensemble des fréquences de raies spectrales (LSF, Line Spectral Frequencies) et le nombre de passage par zéro.
Si le numéro de trame est inférieur à un nombre prédéfini *Ni*, une étape d'initialisation des moyennes à long terme intervient. Dans ce cas précis, la décision de détection d'activité vocale est forcée à 1 si l'énergie de trame obtenue à partir de l'analyse de codage de prédiction linéaire (LPC, Linear Prediction Coding) est supérieure à 15 dB [11]. Sinon, la décision de détection d'activité vocale est forcée à 0. Si le numéro de trame est égal à *Ni*, une étape d'initialisation pour les énergies caractéristiques du bruit de fond intervient.
Lors de l'étape suivante, un ensemble de paramètres de différence est calculé. Cet ensemble est produit par une mesure de différence entre les paramètres de la trame courante et les moyennes glissantes des caractéristiques de bruit de fond. Quatre mesures de différence sont calculées:

- une distorsion spectrale;
- une différence d'énergie;
- une différence d'énergie dans la bande de fréquences basses;
- une différence des nombres de passage par zéro.

La prise de décision initiale sur l'activité vocale est prise lors de l'étape suivante, en utilisant des régions de décision à frontières multiples dans l'espace des quatre mesures de différence. La prise de décision sur l'activité vocale s'effectue à partir de la réunion des régions de décision et la décision de non-activité vocale est le complément logique. La prise en compte de l'énergie, de même que les décisions antérieures sur les trames voisines, sont utilisées pour le lissage de la décision. Les moyennes glissantes doivent être mises à jour uniquement en présence de bruit de fond et non en présence de parole. Un seuil adaptatif est essayé, et la mise à jour a lieu uniquement si le critère de seuil est atteint.

- **Prétraitement**

L'entrée dans le codeur de signaux vocaux est censée être un signal MIC de 16 bits, deux fonctions de prétraitement sont appliquées avant le processus de codage [11]: i) la normalisation du signal, ii) son filtrage passe-haut.

La normalisation consiste à diviser par un facteur de 2 l'énergie d'entrée afin de diminuer la probabilité de dépassements de capacité dans une réalisation en virgule fixe. Le filtre passe-haut sert de précaution à l'encontre de composantes parasites à basse fréquence. On fait appel à un filtre du deuxième ordre avec section des pôles et section des zéros, dont la fréquence de coupure est de 140 Hz. On combine les deux opérations, de normalisation moitié et de filtrage passe haut, en divisant par 2 les coefficients figurant au numérateur de ce filtre, dont l'équation résultante est donnée par la formule suivante:

$$H_{h1}(z) = \frac{0{,}46363718 - 0{,}92724705z^{-1} + 0{,}46363718z^{-2}}{1 - 1{,}9059465z^{-1} + 0{,}9114024z^{-2}}$$

- **Extraction des paramètres**

  - **Fenêtrage et calcul des auto-corrélations**

Pour chaque trame, un ensemble de paramètres est extrait du signal de parole. Le module d'extraction des paramètres peut être partagé entre le détecteur VAD, le codeur vocal actif et le codeur vocal non actif.

La fenêtre d'analyse, notée LP, se compose de deux parties: La première est une demi-fenêtre de Hamming et la seconde est un quart de période d'une fonction cosinus. Cette fenêtre est donnée par l'équation suivante:

$$w_{lp}(n) = \begin{cases} 0{,}54 - 0{,}46\cos\left(\dfrac{2\pi n}{399}\right) & n = 0,\ldots,199 \\ \cos\left(\dfrac{2\pi\,(n - 200)}{159}\right) & n = 200,\ldots,239 \end{cases}$$

L'analyse LP comporte une exploration de 5 ms, c'est-à-dire qu'il faut 40 échantillons issus de la prochaine trame vocale, cela se traduit par un délai algorithmique supplémentaire de 5 ms au niveau du codeur. La fenêtre d'analyse LP s'applique à 120 échantillons issus de trames vocales passées, à 80 échantillons issus de la trame vocale présente, et à 40 échantillons de la trame vocale future. La procédure de fenêtrage est décrite sur la Figure 2.2 [11]

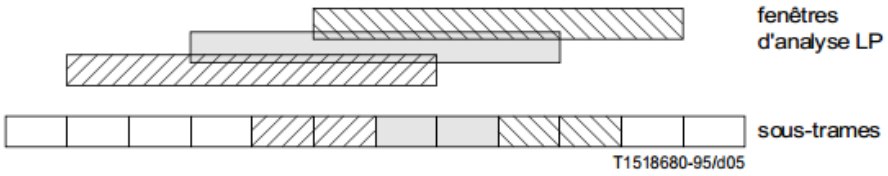


Figure 2.2 : Fenêtrage du VAD G.729

Le signal vocal fenêtré est utilisé pour calculer les coefficients d'autocorrélation.

$$s'(n) = W_{lp}(n)\, s(n) \qquad n = 0, \dots, 239$$

$$r(k) = \sum_{n=k}^{239} s'(n)\, s'(n-k) \qquad k = 0, \dots, 12$$

Pour éviter des problèmes d'ordre arithmétique lors de signaux d'entrée de faible niveau, la valeur de *r(0)* possède une borne inférieure *r(0)* = 1,0. Une extension de la largeur de bande est effectuée sur 60 Hz, par multiplication des coefficients d'autocorrélation comme suit:

$$W_{lag}(k) = exp\left[-\frac{1}{2}\left(\frac{2\pi f_0 k}{f_s}\right)^2\right] \qquad k = 1, \dots, 12$$

où f_0 = 60 Hz est l'extension de largeur de bande et où *fs* = 8000 Hz est la fréquence d'échantillonnage. En outre, le coefficient *r(0)* est multiplié par un facteur de correction du bruit blanc, de 1,0001. Les coefficients d'autocorrélation ainsi modifiés sont donnés par les relations suivantes:

$$r'(0) = 1{,}0001\, r(0)$$

$$r'(k) = W_{lag}(k)\, r(k) \qquad k = 1, \dots, 12$$

- **Fréquences de raies spectrales LSF**

Les coefficients de raies spectrales {LSF} : les coefficients du filtre de prédiction linéaire, les coefficients LPC, sont tout d'abord générés à partir des 13 premiers coefficients d'autocorrélation, ceci grâce à la procédure de Levinson-Durbin et de manière à minimiser l'erreur de prédiction.

Les coefficients d'autocorrélation modifiés, *r'(k)*, sont utilisés pour obtenir les coefficients de filtre LP, a_i, i = 1,....,12, après résolution de l'ensemble d'équations suivant [11] :

$$\sum_{i=1}^{12} a_i r'(|i-k|) = -r'(k) \qquad k = 1, \dots, 12$$

La solution finale est donnée sous la forme $a_j = a_j^{[12]}, j = 0, \dots, 12$ avec $a_0 = 1.0$

Les 13 coefficients LPC sont ensuite convertis en un ensemble de 12 coefficients de raies spectrales $\{LSF\}_{i=1}^{12}$ [11]

$$l_i = \left[\omega_i^{(m)} - \sum_{k=1}^{4} \hat{P}_{i,k}\, \hat{l}_i^{(m-k)} \right] / \left(1 - \sum_{k=1}^{4} \hat{P}_{i,k} \right)$$

- **Énergie dans la pleine bande de fréquences**

L'énergie dans la pleine bande de fréquences E_f est le logarithme du premier coefficient d'autocorrélation normalisé *R(0)*:

$$E_f = 10 \cdot \log_{10}\left[\frac{1}{N} R(0) \right]$$

où N = 240 est la taille de fenêtre d'analyse de codage dans des échantillons de parole.

- **Énergie dans la bande de fréquences basses**

L'énergie dans la bande de fréquences basses E_l mesurée sur la bande de 0 à F l (Hz) est calculée comme suit:

$$E_l = 10 \cdot \log_{10}\left[\frac{1}{N} \mathbf{h}^{\mathbf{T}} \mathbf{R} \mathbf{h} \right]$$

h étant la réponse impulsionnelle d'un filtre FIR dont la fréquence de coupure est de F1 (Hz), ses coefficients sont donnés par [11] :

[-0.0588 -0.0246 0.0554 0.1572 0.2420 1.0000 0.2420 0.1572 0.0554 -0.0246 -0.0588 -0.0473 -0.0106]

R étant la matrice d'autocorrélation de Toeplitz avec les coefficients d'autocorrélation sur chaque diagonale.

- **Nombre de passages par zéro**

Le nombre de passages par zéro ZC normalisé pour chaque trame est calculé par:

$$ZC = \frac{1}{2M}\sum_{i=0}^{M-1}\left[\left|\mathrm{sgn}[x(i)] - \mathrm{sgn}[x(i-1)]\right|\right]$$

où {x(i)} est le signal d'entrée prétraité et M = 80.

- **Initialisation**

En ce qui concerne les N_i premières trames, les paramètres spectraux du bruit de fond, indiqués par $\{\overline{LSF}\}_{i=1}^{12}$ sont initialisés à la moyenne des trames. La moyenne des passages par zéro du bruit de fond, indiquée par *ZC*, est initialisée à la moyenne $\overline{ZC}$ du nombre de passage par zéro des trames.

Les moyennes glissantes de l'énergie de bruit de fond, désignée par $\overline{E_f}$, et l'énergie dans la bande de fréquences basses du bruit de fond, désignée par $\overline{E_l}$, sont initialisées de la façon suivante. Tout d'abord, la procédure d'initialisation utilise la valeur $\overline{E_n}$, définie comme moyenne de l'énergie de trame E_f sur les *Ni* premières trames. Ces trois opérations de moyenne ($\overline{E_n}$, $\overline{ZC}$, $\{\overline{LSF}\}_{i=1}^{12}$) n'incluent que les trames qui ont une énergie E supérieure à 15 dB. En second lieu, la procédure d'initialisation se poursuit comme suit:

si $\overline{En} \le T_1$ alors

$\overline{E}_f = \overline{En} + K_0$

$\overline{E}_l = \overline{En} + K_1$

sinon, si $T_1 < \overline{En} < T_2$ alors

$\overline{E}_f = \overline{En} + K_2$

$\overline{E}_l = \overline{En} + K_3$

sinon

$\overline{E}_f = \overline{En} + K_4$

$\overline{E}_l = \overline{En} + K_5$

Les valeurs des constantes sont données dans le Tableau 2.1 [11]

Le paramètre d'énergie minimale à long terme E_{min} est calculé comme le minimum de E_f sur N_0 trames antérieures. Étant donné que la valeur N_0 est relativement élevée (N_0 = 128), Emin est calculé en utilisant des valeurs enregistrées du minimum E_f sur des segments antérieurs courts.

Tableau des constantes

Nom	Constante	Nom	Constante
N_i	32	N_1	4
N_0	128	N_2	10
K_0	0	T_1	671088640
K_1	−53687091	T_2	738197504
K_2	−67108864	T_3	26843546
K_3	−93952410	T_4	40265318
K_4	−134217728	T_5	40265318
K_5	−161061274	T_6	40265318
a_1	23488	b_1	28521
a_2	−30504	b_2	19446
a_3	−32768	b_3	−32768
a_4	26214	b_4	−19661
a_5	0	b_5	−30802
a_6	28160	b_6	−19661
a_7	0	b_7	30199
a_8	16384	b_8	−22938
a_9	−19065	b_9	−31576
a_{10}	0	b_{10}	−17367
a_{11}	22400	b_{11}	−27034
a_{12}	30427	b_{12}	29959
a_{13}	−24576	b_{13}	−29491
a_{14}	23406	b_{14}	−28087

Tableau 2.1: Table des constantes du VAD G.729

➢ **Calcul des paramètres de différence**

Quatre mesures de différence sont calculées à partir des paramètres de la trame courante et des moyennes glissantes du bruit de fond.

- **La distorsion spectrale ΔS**

La mesure de la distorsion spectrale se calcule par la somme des carrés de la différence entre le vecteur $\{LSF\}_{i=1}^{12}$ de la trame courante et les moyennes glissantes du bruit de fond $\{LSF\}_{i=1}^{12}$.

$$\Delta S = \sum_{i=1}^{p} \left(LSFi - \overline{LSF_i} \right)^2$$

- **Différence d'énergie dans la pleine bande de fréquences ΔE_f**

La mesure de la différence d'énergie dans la pleine bande de fréquences se calcule par la différence entre l'énergie de la trame courante, E_f et la moyenne glissante de l'énergie du bruit de fond, $\overline{E_f}$:

$$\Delta E_f = \overline{E}_f - E_f$$

- **Différence d'énergie dans la bande de fréquences basses ΔE_l**

La mesure de la différence d'énergie dans la bande de fréquences basses se calcule par la différence entre l'énergie dans la bande de fréquences basses de la trame courante E_l , et la moyenne glissante de l'énergie dans la bande de fréquences basses du bruit de fond, $\overline{E_l}$:

$$\Delta E_l = \overline{E}_l - E_l$$

- **Différence des nombres de passages par zéro ΔZC**

La mesure de la différence des nombres de passage par zéro se calcule par la différence entre le nombre de passages par zéro de la trame courante, ZC, et la moyenne courante du nombre de passages par zéro du bruit de fond, $\overline{ZC}$:

$$\Delta ZC = \overline{ZC} - ZC$$

➢ **Décision initiale**

La prise de décision initiale sur l'activité vocale est indiquée par la variable I_{VD} [11], et mise à la valeur 0 ("FAUX") si le vecteur des paramètres de différence se situe dans la zone de non-activité vocale. Sinon, la décision initiale de détection d'activité vocale

est mise à 1 ("VRAI"). Les quatorze frontières de décision dans l'espace quadridimensionnel sont définies comme suit:

1) $si\ \Delta S > a_1 \cdot \Delta ZC + b_1\ alors\ I_{VD} = 1$
2) $si\ \Delta S > a_2 \cdot \Delta ZC + b_2\ alors\ I_{VD} = 1$
3) $si\ \Delta E_f < a_3 \cdot \Delta ZC + b_3\ alors\ I_{VD} = 1$
4) $si\ \Delta E_f < a_4 \cdot \Delta ZC + b_4\ alors\ I_{VD} = 1$
5) $si\ \Delta E_f < b_5\ alors\ I_{VD} = 1$
6) $si\ \Delta E_f < a_6 \cdot \Delta S + b_6\ alors\ I_{VD} = 1$
7) $si\ \Delta S > b_7\ alors\ I_{VD} = 1$
8) $si\ \Delta E_l < a_8 \cdot \Delta ZC + b_8\ alors\ I_{VD} = 1$
9) $si\ \Delta E_l < a_9 \cdot \Delta ZC + b_9\ alors\ I_{VD} = 1$
10) $si\ \Delta E_l < b_{10}\ alors\ I_{VD} = 1$
11) $si\ \Delta E_l < a_{11} \cdot \Delta S + b_{11}\ alors\ I_{VD} = 1$
12) $si\ \Delta E_l > a_{12} \cdot \Delta E_f + b_{12}\ alors\ I_{VD} = 1$
13) $si\ \Delta E_l < a_{13} \cdot \Delta E_f + b_{13}\ alors\ I_{VD} = 1$
14) $si\ \Delta E_l < a_{14} \cdot \Delta E_f + b_{14}\ alors\ I_{VD} = 1$

Si aucun des quatorze états n'est "VRAI", $I_{VD} = 0$

- **Lissage de la décision de détection d'activité vocale**

La décision initiale de détection d'activité vocale est lissée (temps de maintien) pour tenir compte du caractère stationnaire à long terme du signal de parole. Le lissage s'effectue en quatre étapes. Un fanion indiquant que le temps de maintien est intervenu est défini par la variable binaire *v_flag*. Il est mis chaque fois à zéro avant que le lissage de la décision de détection d'activité vocale ne soit effectué. On indiquera la décision d'activité vocale lissée de la trame, de la trame précédente et de la trame antérieure par respectivement S_{VD}^0, S_{VD}^1, S_{VD}^2. Le paramètre S_{VD}^1 est initialisé à 1, et S_{VD}^2 est également initialisé à 1. Pour commencer, $S_{VD}^0 = I_{VD}$. La première étape de lissage est la suivante [11]:

$$\text{si } \left(I_{VD} = 0\right) \text{et } \left(S_{VD}^{-1} = 1\right) \text{et } \left(E > \overline{E}_f + T3\right) \text{alors } S_{VD}^0 = 1 \text{ et } v\_flag = 1$$

Pour la seconde étape de lissage, on définit un paramètre booléen F_{VD}^{-1} et un compteur de lissage C_e . F_{VD}^{-1}.

Le paramètreF_{VD}^{-1} est initialisé à 1 et C_e est initialisé à 0. On indiquera l'énergie de la trame précédente par E_{-1}. La seconde étape de lissage est:

si $\left(F_{VD}^{-1}=1\right)$ et $\left(I_{VD}=0\right)$ et $\left(S_{VD}^{-1}=1\right)$ et $\left(S_{VD}^{-2}=1\right)$ et $\left(\left|E_f - E_{-1}\right| \le T_4\right)\{$

$S_{VD}^{0}=1$

$v\_flag=1$

$C_e = C_e + 1$

si $\left(C_e \le N_1\right)$ {

$F_{VD}^{-1}=1$

}

sinon {

$F_{VD}^{-1}=0$

$C_e=0$

}

}

sinon

$F_{VD}^{-1}=1$

Pour la troisième étape de lissage, on définit un compteur de continuité de bruit C_s qui est initialisé à 0. Si $S_{VD}^{0}=0$, alors le compteur C_s est incrémenté. La troisième étape de lissage est la suivante:

si $\left(S_{VD}^{0}=1\right)$ et $\left(C_s > N_2\right)$ et $\left(E_f - E_{-1} \le T_5\right)$ {

$S_{VD}^{0}=0$

$C_s=0$

}

si $\left(S_{VD}^{0}=1\right) C_s=0$

- **Mise à jour des moyennes glissantes**

Les moyennes glissantes des caractéristiques du bruit de fond sont mises à jour pendant la dernière étape du module VAD. Pendant cette étape, l'état suivant est testé et la mise à jour a lieu si la condition suivante est satisfaite [11]:

$$\text{si } \left(E_f < \overline{E}_f + T_6\right) \text{ alors mettre à jour.}$$

Les moyennes glissantes des caractéristiques de bruit de fond sont mises à jour en utilisant un algorithme autorégressif (AR) du premier ordre. Différents coefficients AR sont utilisés pour différents paramètres et différents ensembles de coefficients sont utilisés au début de l'enregistrement/la conversation ou quand un changement important des caractéristiques de bruit est détecté.

Soit β_{E_f} le coefficient AR pour la mise à jour de E_f , soit β_{E_l} le coefficient AR pour la mise à jour de E_l , soit β_{ZC} le coefficient AR pour la mise à jour de ZC et soit β_{LSF} le coefficient AR pour la mise à jour de $\{LSF\}_{i=1}^{12}$. Le nombre total de trames pour lesquelles la condition de mise à jour a été satisfaite est compté par C_n. Un ensemble différent de coefficients β_{E_f}, β_{E_l}, β_{ZC} et β_{LSF} est utilisé selon la valeur de C_n. La mise à jour AR est effectuée selon:

$$\overline{E}_f = \beta_{E_f} \cdot \overline{E}_f + \left(1-\beta_{E_f}\right) \cdot E_f$$

$$\overline{E}_l = \beta_{E_l} \cdot \overline{E}_l + \left(1-\beta_{E_l}\right) \cdot E_l$$

$$\overline{ZC} = \beta_{ZC} \cdot \overline{ZC} + \left(1-\beta_{ZC}\right) \cdot ZC$$

$$\overline{LSF}_i = \beta_{LSF} \cdot \overline{LSF}_i + \left(1-\beta_{LSF}\right) \cdot LSF_i \quad i = 1,\ldots, p$$

$\overline{E}_f$ et C_n sont en outre mis à jour selon:

si $\left(\text{comptage de trame} > N_0\right)$ et $\left(\overline{E}_f < E_{\min}\right)\{$

$\overline{E}_f = E_{\min}$

$C_n = 0$

$\}$

2.2.2 Adaptive multi-rate (AMR)

En 1998, ETSI [13] a proposé les standards pour les deux options de détection d'activité vocale du codeur de parole à multi-taux adaptatif (AMR). L'AMR a été développé pour le GSM et pour les systèmes de communications mobiles de troisième génération. Ces deux VAD, que nous appellerons par la suite AMRl et AMR2, différent à la fois par leur approche et par leur complexité.

La première méthode, 1'AMR1, peut être schématisée par la Figure 2.3 :

- **Détection du pitch et détection des tonalités :**

Dans leurs travaux, Vahatalo et Johansson [14] expliquent l'utilité de ces modules comme suit. Ces deux détecteurs vont identifier la présence de signaux stationnaires, comme les sons voisés et les tonalités d'information. Il est important de détecter séparément ces signaux stationnaires car la mise à jour de l'estimé du bruit ne sera pas effectuée de la même manière lorsqu'ils sont présents. Il est nécessaire d'utiliser ces deux détecteurs conjointement car chacun compense les faiblesses de l'autre.

En effet, le détecteur du pitch n'identifie pas correctement les signaux ayant plus d'une fréquence fondamentale. Celui des tonalités les détecte. À l'inverse, le détecteur de

tonalités identifie les sons voisés seulement quand le SNR est élevé. Celui du pitch est, lui, capable de les détecter quand le SNR est plus faible.

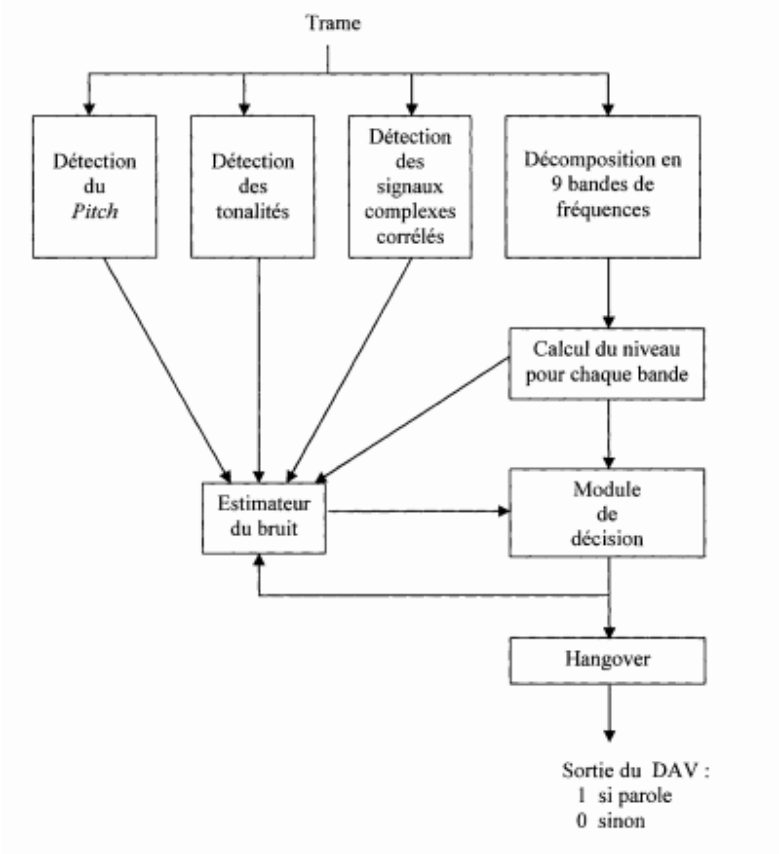


Figure 2.3 : Schéma fonctionnel du VAD AMR1

- **Détection des signaux complexes corrélés:**

Comme il a été mentionné dans le chapitre 1, les codeurs de parole transmettent un bruit de confort lorsque la sortie du VAD est nulle. Selon ETSI [13], en présence de signaux complexes, comme la musique, cette opération risque d'être gênante pour la conversation. Elle ne doit donc pas être effectuée. Pour cela, un détecteur de signaux complexes corrélés est nécessaire. Il consiste à filtrer le signal d'entrée par un filtre passe-haut. Si la sortie du filtre contient des hautes valeurs de corrélation alors un signal complexe corrélé est détecté. Cela repose sur 1'hypothèse que les signaux musicaux possèdent des harmoniques même dans les hautes fréquences alors que le bruit, en général, n'en a que dans les basses fréquences (Vahatalo et Johansson [14]). La description des trois étapes suivantes fait référence aux standards d'ETSI [13]

- **Détection des signaux complexes corrélés :**

Afin de pouvoir extraire les caractéristiques utilisées pour la prise de décision, la trame étudiée est décomposée en 9 bandes de fréquences à 1'aide du banc de filtres illustré par la Figure 2.4. Chaque bloc divise son entrée en une partie passe-haut et une passe-bas et effectue aussi une décimation par 2. D'après Vahatalo et Johansson [14], la décomposition en plus de 9 bandes ne procure pas d'amélioration significative et augmente la complexité.

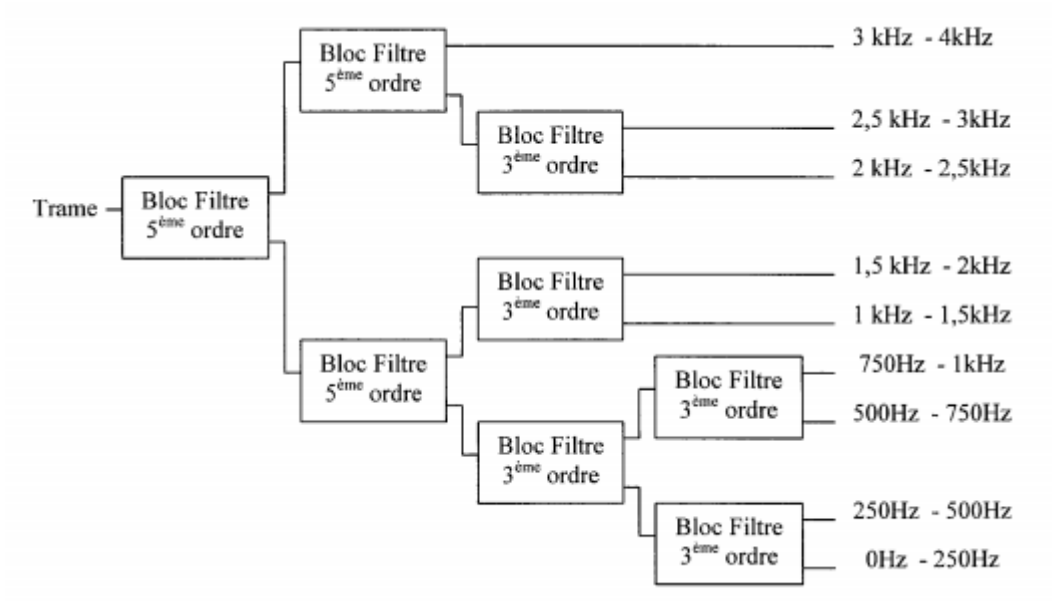


Figure 2.4 : Banc de filtres utilisé par l'AMR1

- **Calcul du niveau pour chaque bande :**

Le niveau du signal est calculé pour chaque bande [13]

$$niveau(k) = \sum |x_k|$$

Avec :

k : la bande de fréquences

x_k : le signal de la bande k

- **Module de décision :**

Comme pour la méthode précédente, un système d'estimation du bruit a été mis en place. À l'aide de ce dernier, la différence entre les niveaux de la trame étudiée et ceux de 1'estimé du bruit est déterminée [13] :

$$som\_rsb = \sum_{k=1}^{9} max\ \left(1, \frac{niveau(k)}{estimé\_bruit(k)}\right)^2$$

La valeur ainsi calculée est ensuite comparée à un seuil et la décision préliminaire est obtenue:

Si som_rsb > seuil alors la trame est ACTIVE

Sinon la trame est INACTIVE

Il est à noter que le seuil utilisé par la règle de décision est adaptatif et dépend du niveau de bruit dans chaque bande.

- **Hangover**

De même que pour le G.729.B, un module de Hangover est utilisé pour corriger la décision préliminaire, si cela est nécessaire. Son principe est le même sauf que la longueur de la période de Hangover est ici variable et dépend du niveau de l'estimé du bruit (Vahatalo et Johansson [14]). De plus, ce module a un but supplémentaire : laisser passer les signaux complexes corrélés. Si le détecteur indique la présence d'un signal complexe corrélé, la sortie du VAD va être forcée à l'état PAROLE afin d'empêcher le déclenchement du générateur de bruit de confort utilisé à la suite du VAD [13].

- **Estimateur du bruit**

La mise à jour de l'estimé du bruit engendre un retard d'une trame car elle s'effectue selon les niveaux d'amplitude de la trame précédente. Ceci a pour but de faciliter la détection des débuts de bouffées de parole, ETSI [13]. Vahatalo et Johansson [14] expliquent le fonctionnement de l'estimateur du bruit de la manière suivante. Si la décision préliminaire est PAROLE ou si le pitch ou une tonalité est détectée, l'estimé du bruit sera revu à la baisse. Dans le cas contraire, il sera revu à la hausse. L'estimé du bruit est réajusté dans chacune des 9 bandes de fréquences de manière indépendante et la vitesse de mise à jour est aussi propre à chacune d'entre elles. Il est à noter que cette vitesse est plus élevée lors d'une revue à la baisse. L'estimation du bruit est régie par l'équation suivante [14]:

$$bruit_{j+1}[k] = (1 - a) \times bruit_j[k] + a \times niveau_{j-1}[k]$$

Avec :

k : la bande de fréquences

j : l'indice de la trame

Comme pour le G.729, un mécanisme pour faire face à une augmentation importante et soudaine du bruit a été mis en place. En effet, dans ce cas la sortie du VAD serait à l'état PAROLE et l'estimé du bruit serait continuellement revue à la baisse. Pour éviter cela, l'augmentation de l'estimé du bruit est permise lorsque la décision préliminaire est PAROLE pendant une période suffisamment longue et lorsque le spectre du signal est stationnaire.

La seconde méthode, celle de l'AMR2, est beaucoup plus complexe et nous ne la décrirons ici que très sommairement [13].

La trame étudiée est séparée en deux sous trames. Chacune d'entre elles subit alors la même procédure, illustrée par la Figure 2.5.

Le signal d'entrée est d'abord converti dans le domaine des fréquences grâce à la transformée de Fourier. Les bandes de fréquences sont regroupées en N_c canaux. Les énergies de chacun d'entre eux sont déterminées. Une fois encore, un estimateur de bruit a été mis au point et permet de calculer les rapports signal à bruit pour les N_c canaux.

Les métriques vocales sont déduites de ces SNR grâce à une fonction non-linéaire. Une décision préliminaire est obtenue pour chaque sous-trame en comparant la somme des métriques vocales à un seuil adaptatif dépendant du bruit. Finalement, la trame entière est dite active si au moins l'une des deux sous-trames est elle-même active.

Les autres blocs sont utilisés pour la mise à jour du bruit. L'estimateur de la déviation spectrale et le détecteur des signaux sinusoïdaux ont pour but d'éviter les réajustements inadéquats de l'estimé du bruit [14]. Ils génèrent des drapeaux unitaires s'il y a détection respective d'une trop grande déviation ou d'un signal sinusoïdal. Ces derniers sont utilisés pour déterminer si une mise à jour de l'estimé du bruit est nécessaire. Si c'est le cas, 1'estimateur du bruit de fond réajuste les caractéristiques du bruit.

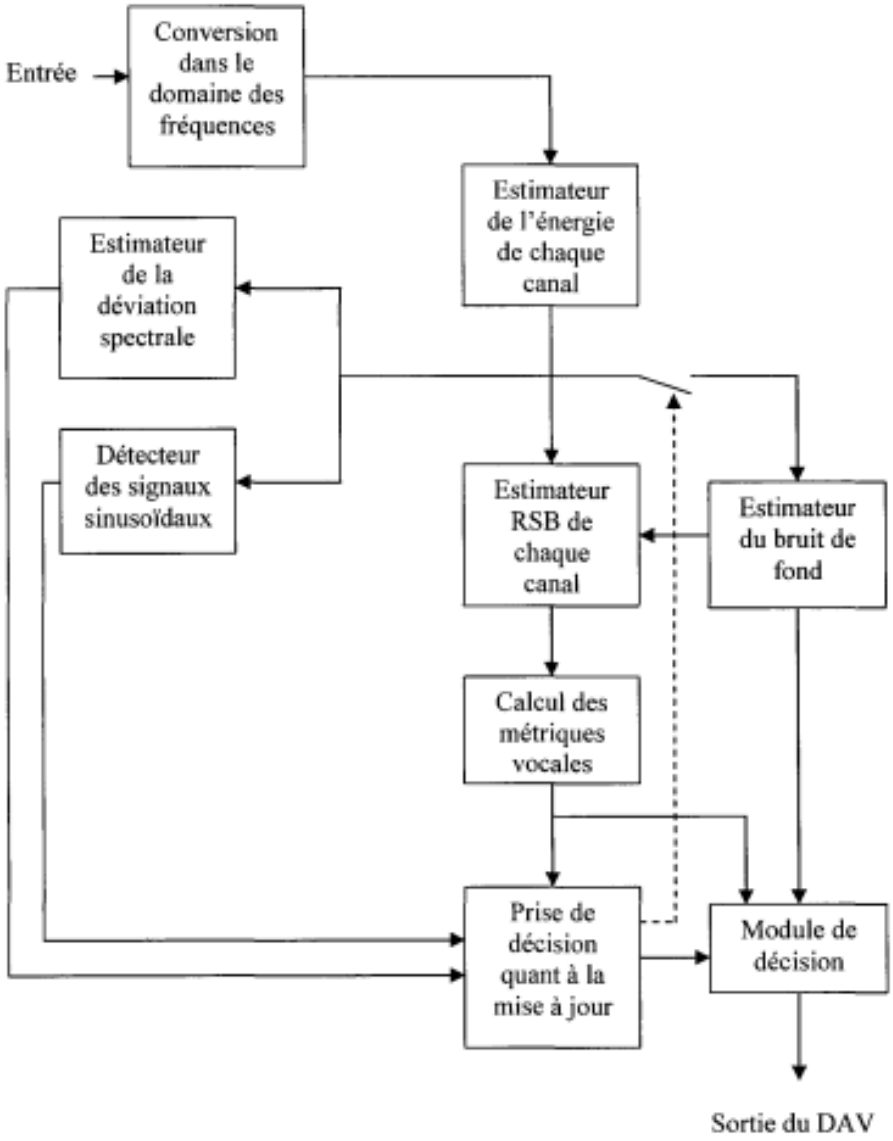


Figure 2.5 : Schéma fonctionnel du VAD AMR2

Les deux VAD présentés ici utilisent une décomposition en bandes de fréquences sur lesquelles ils extraient ensuite les caractéristiques utiles à la prise de décision. Cette approche est très intéressante car elle repose sur le fait que le bruit seul et la parole bruitée ne se répartissent pas de la même manière sur les plages de fréquences. L'AMRl est plus complexe que le G729.B mais il a l'avantage de ré-estimer les caractéristiques du bruit continuellement. L'AMR2, quant à lui, est bien plus lourd d'un point de vue computationnel. La manière de déterminer si une mise à jour de l'estimé du bruit est nécessaire est beaucoup plus élaborée.

2.2.3 Advanced front-end (AFE)

Les performances des systèmes de reconnaissance vocale recevant de la parole transmise sur des canaux mobiles peuvent être considérablement dégradées par rapport à l'utilisation d'un signal non modifié. Le codec ETSI AFE [15] a été conçu pour fonctionner en tant que partie d'un système de reconnaissance vocale distribuée (DSR), dans lequel un canal de données protégé contre les erreurs est utilisé en parallèle avec le canal de signal vocal, pour envoyer une représentation paramétrée de la parole, qui convient à la reconnaissance. AFE comprend deux VAD et un bloc fonctionnel de classification vocale :

- ***VADNest*** est une estimation de bruit dont la sortie est utilisée pour la réduction du bruit via la procédure de filtrage de Wiener. *VADnest* s'applique sur des trames de 10 ms et utilise l'énergie de la trame logarithmique pour la détection d'activité vocale.
- ***VADVC*** est un VAD de classification de voix, il utilise une trame de canal calculée pour 23 bancs de filtre Mel, une table de seuil statique et un schéma de maintien pour la détection d'activité vocale. La classification utilise la sortie de *VADVC*, l'énergie de la trame, le signal de la bande supérieure et l'estimation de la fréquence fondamentale (pitch), pour classer une trame. La sortie est l'une des quatre classes de voix: non vocale, non voisée, à voix mixte, à voix complète.

La décision finale pour AFE VAD est définie comme suit :

$$\text{AFE} = \begin{cases} 1 & \text{if output class} \in \{\text{mixed-voiced, fully-voiced}\} \\ 0 & \text{otherwise} \end{cases}$$

2.2.4 SILK

Développé pour l'application Skype VoIP [16], dans la méthode SILK, le signal d'entrée est traité par un VAD pour produire une mesure de l'activité vocale, ainsi que des estimations d'inclinaison spectrale et de signal sur bruit, pour chaque trame. Le VAD utilise une séquence de bancs de filtres demi-bande pour diviser le signal en quatre sous-bandes: [0 - f_s/16], [f_s/16 - f_s/8]; [f_s/8 - f_s/4]; [f_s/4 - f_s/2]. Ici, f_s est la fréquence d'échantillonnage, soit 8, 12, 16 ou 24 kHz. La sous-bande la plus basse, de [0 à f_s/16], est filtrée passe-haut avec un filtre moyen mobile de premier ordre pour réduire l'énergie aux fréquences les plus basses. Pour chaque trame, l'énergie du

signal par sous-bande est calculée. Dans chaque sous-bande, un estimateur de niveau de bruit suit le niveau de bruit de fond, et le SNR est calculée comme le logarithme du rapport de l'énergie au niveau de bruit. En utilisant ces variables intermédiaires, les paramètres suivants sont ensuite calculés pour une utilisation dans l'analyse de pitch VAD et les autres modules SILK [16] :

- Niveau d'activité de la parole : basé sur le SNR moyen et une moyenne pondérée des énergies des sous-bandes.
- SNR moyen : La moyenne des valeurs SNR de sous-bande.
- SNR sous-bandes lissés : Les valeurs du SNR de sous-bande qui sont lissée temporairement.
- Inclinaison spectrale : Une moyenne pondérée des SNR des sous-bandes, avec des pondérations positives pour les sous-bandes basses et des pondérations négatives pour les sous-bandes. Le signal d'entrée est filtré par un filtre passe-haut pour supprimer la partie la plus basse du spectre qui contient peu d'énergie vocale et peut contenir du bruit de fond. Enfin, le signal est traité par l'estimateur du pitch en boucle ouverte, bien que SILK autorise un signal d'entrée haute fréquence, l'analyse de hauteur fonctionne sur des signaux sous-échantillonnés à 4 et 8 kHz. Ceci est fait en ordre pour réduire la complexité de calcul.

Chapitre 3:
Technique VAD proposée

Résumé

Dans ce chapitre, nous présentons l'état de l'art des méthodes VAD basées sur les décisions statistiques, par la suite, nous décrivons notre approche VAD qui est basée sur le maintien d'un taux de fausse acceptation constant (Constant False Acceptance Rate). La méthode VAD proposée utilise le principe du seuillage adaptatif afin de faire face aux changements du niveau de puissance du bruit de fond dans un environnement non stationnaire. Notre algorithme est comparé au VAD du standard G.729 dans des environnements stationnaires et non stationnaires, moyennant la base de données audio NOIZEUS ainsi que des signaux réalistes enregistrés et noyés dans un bruit de fond expérimental.

3.1 Introduction

Dans ce chapitre, nous proposons une contribution pour la détection de l'activité vocale basée sur les techniques de décision statistiques. Conventionnellement, la théorie de la décision binaire s'appuie sur un test statistique impliquant une opération de seuillage. En traitement VAD, l'utilisation d'un seuillage adaptatif permet de contrôler le taux de fausse acceptation (False Acceptance Rate) qui s'apparente à la probabilité de déclarer une trame comme région vocale, alors que celle-ci provient d'un bruit de fond.

La technique proposée dans ce cadre utilise un seuil dynamique intimement lié à la puissance du bruit local. Des tests séquentiels basés sur l'énergie en pleine bande ont été mis en œuvre afin de rejeter ou d'accepter la trame en cours d'investigation en tant que région vocale active.

La principale caractéristique de l'algorithme proposée réside dans sa capacité à mettre à jour, de façon dynamique, l'estimateur du niveau de bruit en fonction de l'environnement ambiant. En tenant compte de la stationnarité à long terme du signal de parole, nous avons également développé une procédure de lissage qui analyse les dernières décisions partielles ; le but étant de générer le VAD final relatif à la trame sous test. Les performances de l'approche proposée ont été évaluées puis comparées à la norme VAD G.729 dans plusieurs situations, y compris en présence de divers bruits acoustiques environnementaux avec différents SNR. L'analyse des résultats a été réalisée en utilisant la base de données expérimentale NOIZEUS ainsi que plusieurs signaux vocaux, enregistrés dans des situations réalistes.

3.2 VAD basées sur les décisions statistiques

La quasi-totalité des techniques VAD basées sur la théorie de la décision statistique binaire sont fondées sur la maximisation d'un rapport de vraisemblance (Likelihood Ratio Test ou LRT), appliqué à deux hypothèses statistiques communément appelés H_1 et H_0 et faisant référence, respectivement, à l'absence ou à la présence d'une trame vocale. De ce fait, la recherche d'un modèle probabilistique décrivant les variations du signal vocal noyé dans un bruit de fond a de tout temps été considérée comme un thème incontournable dans ce domaine.

- **Etat de l'art**

La majorité des approches VAD basées sur la modélisation statistique du signal vocal noyé dans un bruit de fond ont été inspirées des travaux d'Ephraim et Malah [17] qui ont utilisé ce concept dans le cadre du rehaussement du signal parole (Speech Enhancement). Ensuite, Sohn

et al ont appliqué un modèle statistique Gaussien [18] en vue de l'estimation des paramètres pertinents du signal vocal au moyen de l'approche dite Décision Dirigée (DD) [19]. Par ailleurs, la plupart des algorithmes VAD qui fonctionnent principalement dans le domaine de la transformée de Fourier discrète (DFT) supposent que les spectres du signal parole et du bruit sont distribués selon des lois Gaussiennes. Il a été également établi que les coefficients DFT de parole et du bruit de fond sont efficacement modélisés par des fonctions de densité de probabilité (pdfs) telles que les distributions Gamma et Laplace [20], [22].

L'un des principaux problèmes du VAD est l'estimation robuste du rapport signal sur bruit (SNR) et, dans ce cadre, de nombreuses études ont été réalisées. En effet, dans [19] le schéma d'estimation DD a été appliqué pour calculer le SNR a priori ; cette technique a été initialement proposée dans [17] pour réduire le biais, engendré par l'estimation du maximum de vraisemblance (ML) de l'amplitude spectrale. La méthode DD s'est avérée fournir une estimation plus lissée (Smoothed) pour le SNR que l'approche ML pendant les périodes de silence [19]. Le comportement général ainsi que tous les détails de l'approche DD ont été analysés et sont disponibles dans la référence [19]. Une autre innovation de la VAD consiste en l'incorporation de l'opération de lissage des décisions générées dans l'estimation du spectre de puissance afin d'éviter des phénomènes de hachage dans les intervalles voisés [22] - [23].

S'agissant de la modélisation statistique du signal vocal, plusieurs modèles « candidats » pour caractériser la distribution des composantes spectrales dans divers environnements bruités, ont été étudiés [24]. Afin de décrire la distribution des coefficients DFT, la pdf Gaussienne traditionnelle ainsi que les pdfs complexes Laplaciennes et Gamma ont été appliquées. Chaque modèle paramétrique est évalué à travers un test de la qualité de l'approximation qui mesure l'écart entre le modèle considéré et la distribution empirique, dans cette étude le test de validation du modèle qui a été utilisé est celui de Kolmogorov – Smirnov (KS) [25].

Sur la base de l'analyse des tests KS, il a été constaté que les modèles Laplacien et Gamma sont les plus adaptés pour modéliser les spectres de parole dans diverses situations bruitées. En fait, la robustesse du modèle mixte (Gauss plus Laplace) explique l'utilisation fréquente de ce formalisme dans la plupart des articles traitant de cette thématique. Nous présentons dans ce qui suit, les grands traits de cette modélisation :

Tout au long de cette section, nous considérons deux fonctions de densité de probabilité distinctes (pdfs) qui représentent la voix et les distributions d'amplitude de bruit du modèle proposé.

On suppose que les deux distributions pour la parole et le bruit sont respectivement les lois «Laplacienne» et «Gaussienne», comme établis dans [26] où différents modèles de distribution de la parole sont présentés dans la Figure 3.1.

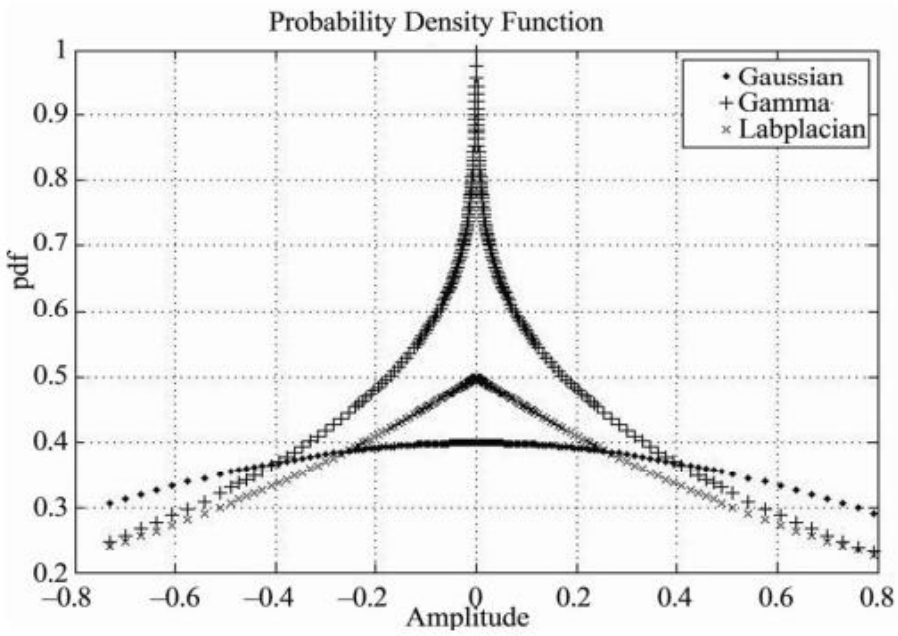


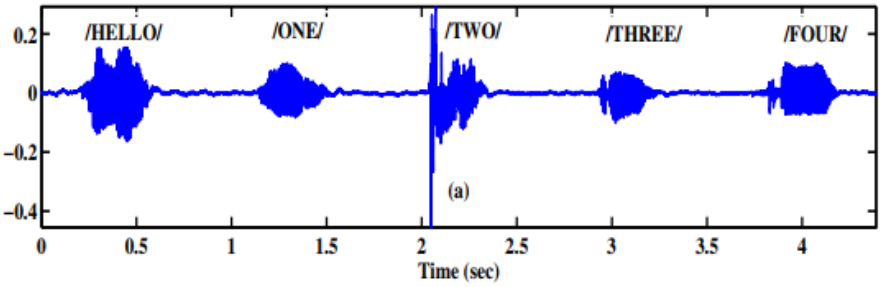
Figure 3.1 Distributions de la parole.

En supposant que le bruit de fond est du type AWGN, le signal vocal bruité est alors représenté comme suit :

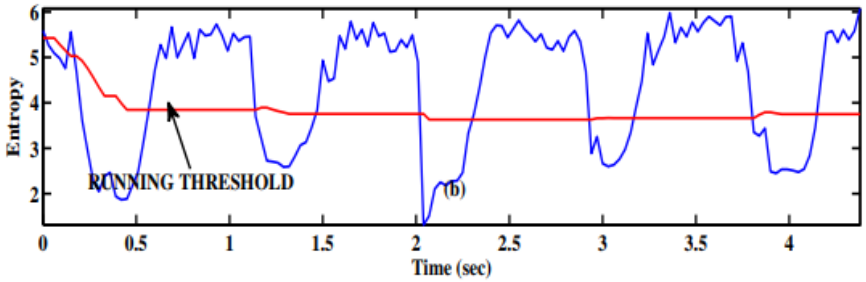
$Y = X_{signal} + N_{AWGN}$, où Y désigne la parole bruitée, X_{signal} correspond au signal parole claire et N_{AWGN} représente le bruit. Notant que ce modèle considère que X_{signal} et N_{AWGN} sont des processus statistiquement indépendants l'un de l'autre.

Dans [27], R. Muralishankar *et al*, ont introduit une nouvelle technique pour différencier les intervalles d'activité vocale des régions silencieuses au niveau d'un flux vocal. Selon [27], cette méthode semble convenir parfaitement à la transmission de la voix sur Internet VoIP.

Dans un objectif de différenciation entre les zones de silence et les zones de parole, une mesure d'entropie, basée sur les variables d'écart des Statistiques d'Ordre (OS) , a été utilisée. Dans ce cadre, les auteurs ont mis au point un algorithme utilisant un seuil adaptatif, également appelé seuil glissant (Running Threshold) (Figure 3.2) pour minimiser les erreurs de détection VAD. Les performances de cette approche sont comparées au VAD intégré dans le Codec AMR [12]. Il a été établi que cette approche permet une économie substantielle de la bande passante tout en maintenant une qualité acceptable des flux vocaux. En outre, l'approche proposée a amélioré la probabilité de détection de l'activité vocale par rapport aux schémas AMR dans des environnements fortement bruités.



(a) Forme d'onde de parole d'origine



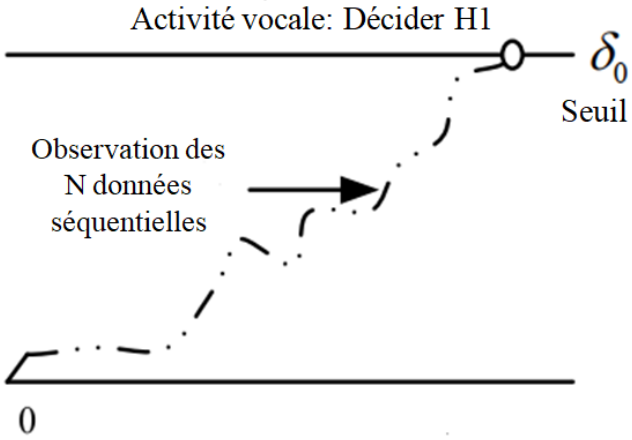
(b) Courbe d'entropie avec seuillage sous une taille de trame de 20 ms

Figure 3.2 Décisions VAD pour un discours clair

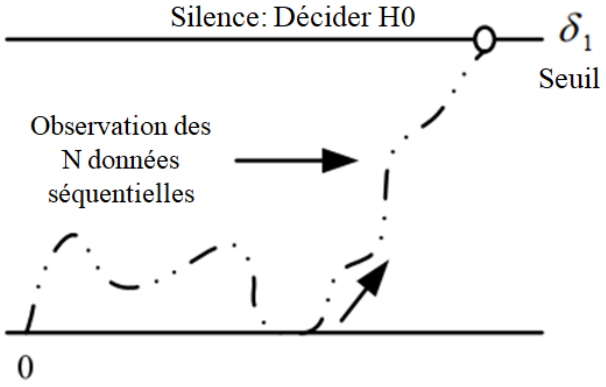
Dans [26], Etellisi et Kazakos ont proposé un nouvel algorithme VAD basé sur un test de changement des séquences de détection (SDCT-VAD). L'algorithme suit efficacement les points limites entre l'activité vocale continue et les périodes de silence. Dans ce cas précis, le bruit et la parole bruitée sont modélisés respectivement par une distribution Gaussienne et une distribution mixte Laplacienne plus Gaussienne.

L'algorithme a été testé dans un scénario en temps réel, pour montrer sa robustesse ainsi que sa faible complexité d'implémentation.

L'algorithme ci-dessus fonctionne séquentiellement via l'utilisation de deux seuils prédéfinis et notés respectivement 0 et δ comme indiqué à la Figure 3.3. Un suivi séquentiel est effectué en analysant les transitions entre les intervalles de silence et les régions voisées à travers l'utilisation de 2 seuils prédéfinis. Cette technique permet d'estimer pour chaque échantillon, l'environnement auquel il appartient. En d'autres termes, l'algorithme estime la densité de probabilité à laquelle l'échantillon appartient, parmi les deux pdf possibles ; à savoir la pdf Gaussienne (f_0) ou la pdf mixte Gauss plus Laplace (f_1).



(a) Détecter le changement de bas vers le haut



(b) Détecter le changement de haut vers le bas

Figure 3.3 Principe de la Technique SDCT-VAD

3.3 Méthode VAD proposée

3.3.1 VAD basé sur l'énergie dans la pleine bande de fréquences

La diversité et la nature variable de la voix active et du bruit ambiant rendent le problème VAD assez compliqué en pratique. L'algorithme VAD prend une décision d'activité vocale toutes les 10 ms conformément à la taille de trame du codeur vocal G.729 [11]. L'ensemble des paramètres couramment utilisés pour générer la décision VAD se compose de quatre mesures différentes; à savoir, l'énergie pleine bande, l'énergie de la bande basse, le taux de passage par zéro et la mesure spectrale.

Pour les besoins de ce travail, nous donnons seulement l'expression de l'énergie pleine bande par trame, E_{FB}, qui correspond au logarithme du premier coefficient d'autocorrélation normalisé [11]

$$E_{FB} = 10.\log\left[\frac{1}{N}R_x(0)\right] \quad (1)$$

où N est la taille de la fenêtre d'analyse exprimée en échantillons de parole, $Rx(0)$ est le coefficient d'autocorrélation et x_j sont les amplitudes échantillonnées du signal audio, c'est-à-dire

$$R_x(0) = \sum_{j=1}^{N} x_j^2 \quad (2)$$

Dans le travail proposé, nous considérons que les amplitudes de la parole et du bruit additif sont distribuées respectivement selon les modèles Laplacien et Gaussien largement utilisés [26].

De plus, pendant les intervalles non vocaux (hypothèse H_0), les amplitudes du bruit gaussien blanc additif (AWGN) sont supposées être des variables aléatoires indépendantes et identiquement distribuées (IID). En d'autres termes, les amplitudes non voisées sont régies par la fonction de densité de probabilité Gaussienne centrée (pdf) avec l'écart type $(\sigma/2)$, également noté $G(0, \sigma/2)$.

Le problème abordé étant basé sur l'énergie comme variable de décision, nous déterminons dans ce qui suit la pdf de la variable aléatoire liée à l'énergie de la trame sous investigation. Dans ce qui suit, nous utilisons le premier coefficient de l'autocorrélation comme variable de décision au lieu de l'énergie pleine bande. En effet, puisque l'équation (1) se réfère à une fonction monotone croissante, l'énergie pleine bande et l'autocorrélation sont des variables aléatoires équivalentes en terme de test d'hypothèse statistique. À cette fin, nous définissons la variable de décision pour la $k^{ème}$ trame de longueur $2M$ échantillons comme

$$E_k = \sum_{i=1}^{2M} x_i^2 \quad (3)$$

Où, les variables aléatoires IID *xi {i = 1,…, 2M}* correspondent aux amplitudes AWGN pendant une trame non voisée. En prenant la variable aléatoire y_i comme la somme de deux variables aléatoires gaussiennes au carré, c'est-à-dire $y_i = x_{2i-1}^2 + x_{2i}^2$,nous pouvons facilement voir que y_i suit une pdf exponentiel de paramètre σ, donc

$$f_{y_i}(y) = \frac{1}{\sigma} exp^{\frac{-y}{\sigma}} \quad ,\{ i = 1, \dots, M \} \quad (4)$$

Après substitution de y_i dans (3), la variable de décision E_k devient la somme de *M* variables aléatoires indépendantes et exponentiellement distribuées. Aussi, sous l'hypothèse H_0, on montre aisément que la pdf de E_k correspond simplement à une distribution Gamma [28] de paramètres *(M, σ)*, ainsi

$$f_{E_k}(\varepsilon) = \frac{\varepsilon^{M-1} \exp(-\frac{\varepsilon}{\sigma})}{\Gamma(M)\sigma^M} \quad (5)$$

Les tests d'hypothèses binaires séquentielles permettent de décider, à chaque étape k, si la k^{ème} trame sous test correspond à une région non voisée (hypothèse nulle H_0) ou contient de la voix noyée dans du bruit (hypothèse alternative H_1). Pour réaliser un tel schéma de décision, nous devons définir le taux de fausse acceptation, noté *FA*, comme étant la probabilité de décider d'une trame voisée pendant un intervalle de silence (H_0), donc

$$F_A = Prob(E_k > Th/H_0) \quad (6)$$

Ensuite, l'énergie E_k de la $k^{ème}$ trame sous test est comparée au seuil adaptatif *Th*, puis une décision partielle d_k est prise en faveur de H_1 (trame voisée) ou H_0 (trame non voisée) selon le test d'hypothèse suivant

$$E_k \gtrless_{H0}^{H1} Th \quad (7)$$

Dans l'approche proposée, le seuil adaptatif est étroitement lié à l'environnement bruité ambiant, il est estimé, à chaque étape, en ne prenant en compte uniquement les énergies des trames précédemment déclarées en tant que trames non vocales (bruit). Le seuil adaptatif peut être défini comme un niveau moyen d'énergie locale, donc

$$Th = T.Z \quad (8)$$

Où $Z = \sum_{i=1}^{N_0} E_i$ correspond à l'estimation du niveau moyen de l'énergie de fond et les E_i sont les énergies des trames qui ont été précédemment décidées comme non voisées. Le facteur

d'échelle *T* est choisi pour maintenir un taux nominal de fausse acceptation (*FA*) désiré sous l'hypothèse nulle H_0 (absence d'activité vocale).

3.3.2 Taux de fausse acceptation constant

Afin de calculer le facteur d'échelle *T* qui maintient un taux nominal de fausse acceptation, nous devons procéder à un développement mathématique pour l'obtention d'une expression compacte du taux de fausse acceptation *FA*. Ainsi, en substituant *Th* dans (6) et en utilisant les pdf f_Z et f_{Ek} des variables aléatoires E_k et *Z* respectivement, l'équation (6) devient

$$F_A = \int_0^\infty f_z(z) \int_{Th}^\infty f_{E_k}(e) de dz \tag{9}$$

Comme indiqué dans [28], en utilisant les fonctions de génération du moment (MGF) et l'intégrale de contour, *FA* s'écrit sous la forme :

$$F_A = -\sum_j \left(Res(\omega^{-1} \Phi_{E_k}(\omega) \Phi_z(-T\omega)) \right|_{\omega=\omega_j} \tag{10}$$

Où, *Res [.]* désigne le résidu, Φ_Z et Φ_{E_k} sont le MGF de la statistique *Z* et de la variable aléatoire E_k sous l'hypothèse H_0 respectivement. ω_j sont les pôles de $\Phi_{E_k}(\omega)$ situés dans le demi-plan ω-complexe gauche.

Comme indiqué précédemment, la statistique $Z = \sum_{i=1}^{N_0} E_i$ est exprimée comme la somme de N_0 variables aléatoires indépendantes E_i. À son tour, chacune des énergies E_i a été définie comme étant la somme de *M* variables aléatoires indépendantes et exponentiellement distribuées y_j, ce qui est clairement illustré par l'équation (3). Puisque les variables aléatoires y_j sont IID, le MGF de leur somme devient simplement le produit des MGF individuels [28], ce qui donne

$$\Phi_{E_k}(\omega) = \prod_{i=1}^{M} \Phi_{y_i}(\omega) \text{ et } \Phi_z(\omega) = \frac{1}{(1+\omega)^{MN_0}} \tag{13}$$

Où *2M* est le nombre de paires d'échantillons par trame, tandis que N_0 désigne la taille du tampon du bruit conformément à la condition initiale qui stipule que les premières trames de n'importe quelle communication ne contiennent aucune activité vocale (bruit de fond).

En substituant l'équation (13) dans l'équation (10) et en résolvant le résidu au pôle $\omega_0 = -1$ d'ordre de multiplicité *M*, on obtient

$$F_A = \frac{-1}{\Gamma(M)} \left(\frac{1}{(-T)^{MN_0}} \right) \frac{d^{M-1}}{d\omega^{M-1}} \left\{ \omega^{-1} \left(\omega - \frac{1}{T} \right)^{-MN_0} \right\}_{\omega_0=-1} \tag{14}$$

avec $\Gamma(M)=(M-1)!$

Afin de résoudre (14), on prend procède au changement de variable suivant :

$U = \omega^{-1}, V = (\omega - \frac{1}{T})^{-MN_0}$, alors

$$\frac{d^L}{d\omega^L}(U.V) = \sum_{i=0}^{L} \begin{pmatrix} L \\ L-i \end{pmatrix} U^{(L-i)} V^{(i)}$$

Où $U^{(L-i)}$ et $V^{(i)}$ sont les $(L-i)^{ème}$ et $i^{ème}$ fonctions dérivées de U et V, respectivement.

En utilisant (15) dans (14), nous obtenons après quelques simples manipulations algébriques, le taux de fausse acceptation ci après

$$F_A = \sum_{i=0}^{M-1} \binom{MN_0 - 1 + i}{MN_0 - 1} \frac{T^i}{(1+T)^{MN_0+i}} \qquad (16)$$

Notant que le facteur d'échelle qui garanti un taux nominal de fausse acceptation sous l'hypothèse H_0, doit être calculé numériquement à partir de (16). Cette tâche préliminaire est une opération qui s'effectue en temps différé.

3.3.3 Principe de fonctionnement du VAD proposé

Définissons d'abord les paramètres opérationnels les plus importants à utiliser dans l'algorithme VAD proposé. En fait, une fois que le facteur d'échelle pré-calculé T est établi, l'algorithme VAD doit assurer une probabilité d'erreur de test d'hypothèse (FA) constante, à l'intérieur des régions de silence (non voisées). Rappelons que les décisions partielles d_k sont générées avant l'opération de lissage, tandis que le VAD final est fourni après le lissage. La tâche de lissage permet d'accepter un intervalle comme étant Voix Active uniquement si P trames successives, précédant celle en cours, ont été détectées comme voisées ($d_{k-1} = 1$, $d_{k-2} = 1, \ldots, d_{k-P} = 1$). De même, lorsque l'algorithme balaie une région voisée, un intervalle n'est déclaré non voisé que si P trames successives ont été détectées comme du bruit. En fait, les conditions mentionnées ci-dessus sont utilisées pour satisfaire la caractéristique stationnaire de la parole et d'éviter, par conséquent, les discontinuités indésirables. Dans ce qui suit, nous donnons quelques détails sur les principales tâches du schéma VAD proposé, tel qu'illustré sur la Figure 3.6.

Après un échantillonnage à 8 kHz des enregistrements audio, l'algorithme commence par former des trames d'une taille de 10 ms, ce qui correspond à *2M = 80* échantillons. Lors de la phase d'initialisation, les N_0 premières trames (supposées non vocales) sont chargées dans un tampon de bruit du type First In First Out (FIFO) pour initialiser le seuil adaptatif (*Th = TZ*) et pour estimer le niveau de bruit ambiant. Ensuite, à la $k^{ème}$ étape ($k > N_0$), une décision partielle d_k est générée selon (7) pour déclarer si la $k^{ème}$ trame est voisée (H_1) ou non (H_0). Ensuite, un ensemble de décisions partielles d_k est chargé dans un registre binaire à décalage de longueur (*P + 1*). En effet, les P décisions partielles précédant la décision en cours d_k, sont continuellement vérifiées pour effectuer le lissage si certaines conditions particulières sont satisfaites. Comme le montre la Figure 3.6, le registre de décisions partielles correspond aux paramètres d'entrée de la procédure de lissage.

- **Bloc de lissage**

L'objectif du lissage est d'éliminer les discontinuités indésirables aussi bien dans les régions vocales que dans les intervalles de silence. Ceci est réalisé en générant des décisions VAD finales lissées ; par conséquent, la qualité de la détection de la parole sera améliorée. Toute fois, on notera que la sortie du VAD sera retardée de *P* trames, ce qui induit une latence insignifiante de quelques dizaines de millisecondes. En tenant compte de la faible dynamique du signal vocal, le retard obtenu ne présente aucun impact sur le mode de fonctionnement en temps réel du système proposé.

Le lissage est effectué en inversant les décisions partielles de quelques trames (parmi *P* trames successives) qui ont été déclarées non voisées dans les intervalles de parole. En effet, lorsqu'une transition d'intervalle de silence vers une région de parole est détectée, au moins *P* trames successives doivent être détectées comme voisées pour prétendre être classées comme parole (*VAD = 1*). De même, lors du passage de la parole vers une région bruitée, une décision est prise en faveur de l'hypothèse H_0 (*VAD = 0*), si au moins *P* décisions partielles successives non vocales ont été générées. En fait, l'ensemble des décisions d_k peuvent subir ou ne pas subir de lissage, selon que les conditions mentionnées ci-dessus sont remplies. Finalement, la décision finale de VAD relative à la trame sous test générée. En fait, l'objectif principal de cette procédure se résume à éliminer certaines singularités pouvant affecter le caractère stationnaire du signal vocal, ce qui est clairement illustré par les exemples décrivant les 2 cas de Figure possible (Figure 3.4 et Figure 3.5).

Trame N°	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
VAD avant lissage	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VAD après lissage	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VAD retardée					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.4: Lissage dans un intervalle de silence

Trame N°	.....	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
VAD avant lissage	.....	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1	0	0	0	0
VAD après lissage	.....	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
VAD retardée	.....					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 3.5: Lissage dans un intervalle de parole

S'agissant du premier scénario (Figure 3.4) ; on remarque que la région allant de la trame N° 1 à la trame N° 41 correspond vraisemblablement à un intervalle de silence. Par conséquent, les trames 22, 23 et 26 paraissent en contradiction avec la dynamique signal, en d'autres termes, il est quasiment irréaliste d'avoir une activité vocale sur une durée aussi courte (20 ms). Ainsi, la procédure de lissage se doit d'éliminer ces singularités en inversant l'état des décisions partielles d_k correspondantes. Toutes fois, on mettra en évidence, la latence qui en résulte de cette opération et qui est de l'ordre de 40 ms ($P+1$ avec $P=3$).

Dans le deuxième scénario (Figure 3.5) ; des trames de silences 84, 85 et 87 de courtes durées (20 ms) et (10 ms) apparaissent dans un intervalle de parole allant de la trame N° 65 à la trame N° 91, la méthode de lissage inverse la décision de ces trames vue que dans la nature de la parole, on peut pas avoir une pause de 20 ms.

- **Mise à jour du tampon de bruit**

L'opération de mise à jour consiste à insérer les énergies non vocales les plus représentatives de l'environnement local. En tenant compte de la sortie du bloc de lissage, à chaque fois que $d_{k-P} = 0$, l'algorithme insère, dans le registre FIFO Noise-Buffer, l'énergie de la trame non voisée précédemment détectée (E_{k-P}). Il est important de noter que le contenu de ce buffer sera ultérieurement utilisé dans le prochain test d'hypothèse.

- **Ajustement du facteur d'échelle**

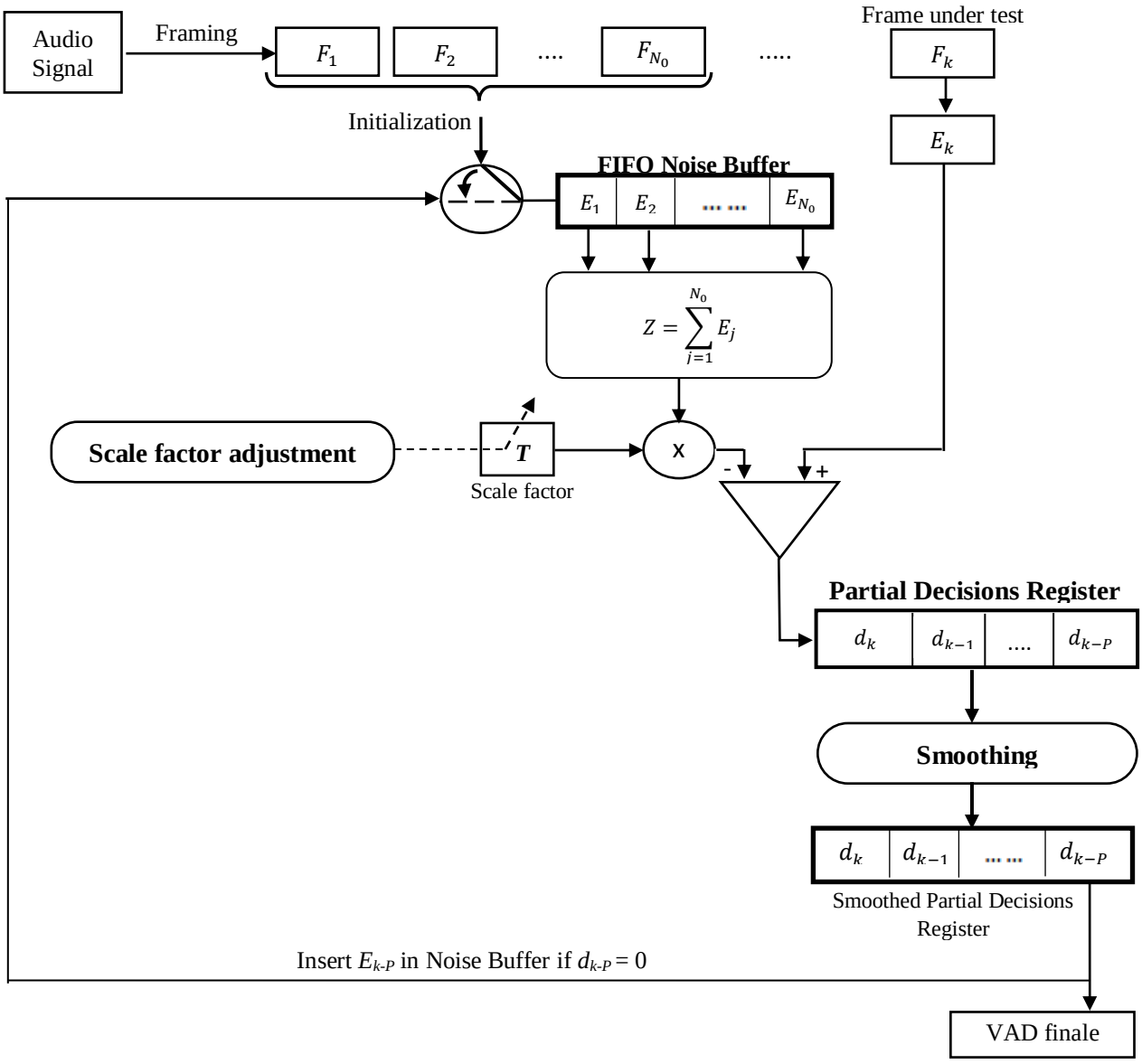
Dans un bruit de fond homogène, où le niveau de puissance est presque constant, le seuil adaptatif proposé $Th = TZ$ assure un taux de fausse acceptation constant grâce au facteur d'échelle pré-calculé T (Equation.16). Cependant, lorsque des transitions de puissance apparaissent dans des environnements bruités non stationnaires, le VAD proposé souffre d'un taux excessif de fausse acceptation, ce qui pourrait avoir un impact négatif sur les performances du système. Pour éviter cet inconvénient, le facteur de mise à l'échelle doit être ajusté lorsque l'algorithme VAD balaie une région de transition (d'un bruit de puissance élevée vers un niveau plus faible ou vice versa) si certaines conditions sont remplies. Comme illustré sur la Figure 3.6.b, un ensemble important de valeurs d'énergies est sauvegardé dans un tampon FIFO contenant f_u trames précédant celle en cours d'investigation. Nous supposons que les valeurs E_{max} et E_{min} représentent respectivement le maximum et le minimum des énergies contenues dans le tampon mentionné. Pour chaque ensemble de f_u trames, la différence glissante d'énergie (E_{max} - E_{min}) est comparée à un seuil d'écart d'énergie prédéfini Δ_E, pour décider si une transition de puissance brusque s'est produite. En règle générale, un écart d'énergie supérieur à 15 dB indique un changement significatif de l'environnement de bruit de fond. Par conséquent, l'ajustement du facteur d'échelle a lieu si la condition suivante est satisfaite

$$E_{max} - E_{min} \geq \Delta_E \qquad (17)$$

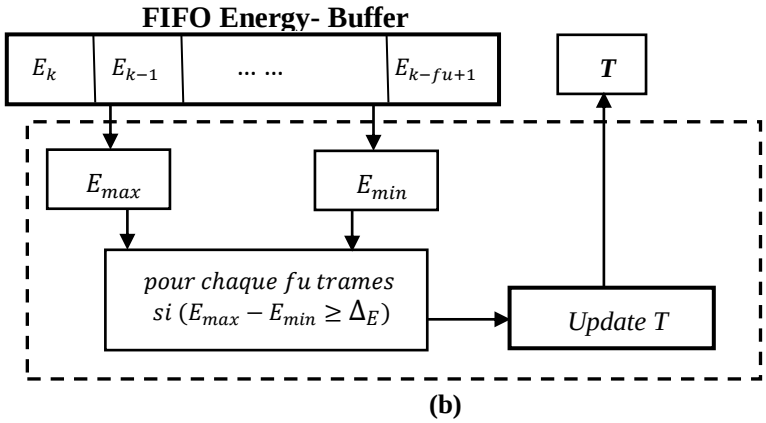
Notant que la condition (17) doit être testée, toutes les f_u trames, afin de réinitialiser le seuil adaptatif TZ dans l'intervalle [E_{min}, E_{max}] à chaque fois qu'un changement d'environnement significatif est détecté; dans ce cas, le facteur d'échelle T est mis à jour comme suit

$$T = \frac{E_{min} + \alpha.\Delta E}{Z} \qquad (18)$$

Où α correspond au paramètre de niveau d'ajustement fixé empiriquement par une simulation, en exploitant intensivement la base de données NOIZEUS [29].



(a) Principe de fonctionnement du VAD proposé



(b)

(b) Ajustement du paramètre d'échelle T

Figure 3.6: Diagramme du VAD proposée

3.4 Résultats et discussion

Les performances de l'algorithme VAD proposé sont évaluées et comparées à celles du VAD standardisé G.729 en utilisant la base de données NOIZEUS [29]. Un bruit blanc synthétique, avec un rapport signal sur bruit (SNR) contrôlable, a été généré et ajouté aux locutions de la base de données NOIZEUS afin d'analyser la méthode VAD, en particulier en présence d'environnements bruités non stationnaires.

3.4.1 Base de données NOIZEUS

- **Description**

Un ensemble de locutions bruyantes (NOIZEUS) a été développé et mis à la disposition des différents groupes de recherche, pour faciliter l'étude des algorithmes de traitement audio. La base de données bruyante contient 30 phrases IEEE (produites par trois hommes et trois femmes) corrompues par huit différents bruits du monde réel à différents niveaux de SNR. Le bruit provient de la base de données AURORA [30] et comprend le bruit des environnements suivants : Passage des trains, babillage, voitures, hall d'exposition, restaurants, rues, aéroports et gares. Cette base de données est mise gratuitement à la disposition des chercheurs [29]

- **Matériel et techniques**

Trente phrases de la base de données de phrases IEEE [31] ont été enregistrées dans une cabine insonorisée à l'aide de l'équipement d'enregistrement Tucker Davis Technologies (TDT). Les phrases ont été prononcées par trois hommes et trois femmes. La base de données IEEE (720 phrases) a été utilisée car elle contient des phrases phonétiquement équilibrées avec une prédictibilité relativement faible du contexte des mots. Les trente phrases ont été sélectionnées dans la base de données IEEE de manière à inclure tous les phonèmes de la langue anglaise américaine. La liste des phrases enregistrées pour NOIZEUS est donnée dans le Tableau 3.1. Les phrases ont été initialement échantillonnées à 25 kHz et sous échantillonnées à 8 kHz.

- **Niveaux de rapport signal sur bruit**

Dans ce travail, sept types de bruit ont été considérés comme précédemment indiqué. Les signaux de bruit sont ajoutés à la parole à un SNR variable de 15 dB, 10 dB, 5 dB et 0 dB. Pour analyser les performances du schéma VAD proposé, nous avons utilisé environ 840 phrases comprenant divers types de bruits à différents SNR. Les fichiers audio nécessaires ont

été enregistrés au format wav avec un codage PCM (Pulse Code Modulation) 16 bits et une fréquence d'échantillonnage de 8 kHz.

Nom de fichier	Locuteur	Genre	Texte
sp01.wav	CH	M	The birch canoe slid on the smooth planks.
sp02.wav	CH	M	He knew the skill of the great young actress.
sp03.wav	CH	M	Her purse was full of useless trash.
sp04.wav	CH	M	Read verse out loud for pleasure.
sp05.wav	CH	M	Wipe the grease off his dirty face.
sp06.wav	DE	M	Men strive but seldom get rich.
sp07.wav	DE	M	We find joy in the simplest things.
sp08.wav	DE	M	Hedge apples may stain your hands green.
sp09.wav	DE	M	Hurdle the pit with the aid of a long pole.
sp10.wav	DE	M	The sky that morning was clear and bright blue.
sp11.wav	JE	F	He wrote down a long list of items.
sp12.wav	JE	F	The drip of the rain made a pleasant sound.
sp13.wav	JE	F	Smoke poured out of every crack.
sp14.wav	JE	F	Hats are worn to tea and not to dinner.
sp15.wav	JE	F	The clothes dried on a thin wooden rack.
sp16.wav	KI	F	The stray cat gave birth to kittens.
sp17.wav	KI	F	The lazy cow lay in the cool grass.
sp18.wav	KI	F	The friendly gang left the drug store.
sp19.wav	KI	F	We talked of the sideshow in the circus.
sp20.wav	KI	F	The set of china hit the floor with a crash.
sp21.wav	SI	M	Clams are small, round, soft and tasty.
sp22.wav	SI	M	The line where the edges join was clean.
sp23.wav	SI	M	Stop whistling and watch the boys march.
sp24.wav	SI	M	A cruise in warm waters in a sleek yacht is fun.
sp25.wav	SI	M	A good book informs of what we ought to know.
sp26.wav	TI	F	She has a smart way of wearing clothes.
sp27.wav	TI	F	Bring your best compass to the third class.
sp28.wav	TI	F	The club rented the rink for the fifth night.
sp29.wav	TI	F	The flint sputtered and lit a pine torch.
sp30.wav	TI	F	Let's all join as we sing the last chorus.

Tableau 3.1: Liste des phrases utilisées dans NOIZEUS

3.4.2 Critères de comparaison et VAD idéal

Pour évaluer la méthode VAD proposée et comparer notre approche avec le standard G.729, les critères objectifs suivants [32] ont été utilisés:

• **Correct:** Décisions correctes prises par le VAD.

• **FEC** (front end clipping): écrêtage dû à la mauvaise classification de la voix comme bruit lors du passage du silence à l'activité vocale ou vice versa.

• **MSC** (mid speech clipping): coupure due à la mauvaise classification de la voix comme bruit pendant une région d'activité vocale.

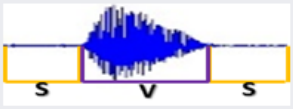
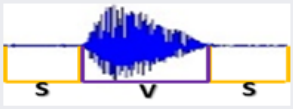
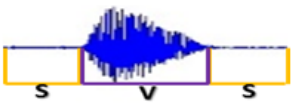
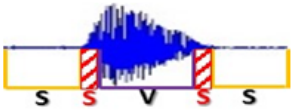
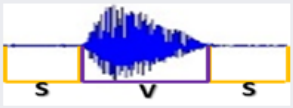
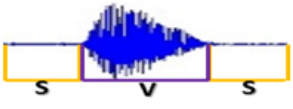
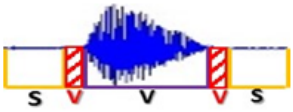
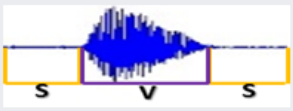
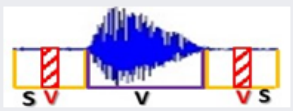
• **OVER** (carry over): bruit interprété comme de la voix en passant de l'activité vocale au bruit ou d'un bruit à l'activité vocale.

• **NDS** (noise detected as speech): bruit interprété comme parole dans une zone de silence.

Tous les critères ci-dessus énumérés sont divisés par le nombre total de trames et donnés en pourcentage (%). La combinaison de FEC et MSC donne le rejet **TR** (True Rejection), tandis que la combinaison de **OVER** et **NDS** fournit la fausse acceptation expérimentale (**FAex**). Le **TR** indique la perte de détection dans les intervalles de parole, tandis que le **FAex** indique le taux réel des régions non vocales classées comme voisées.

La Figure 3.7 illustre les critères de comparaison :

V: voiced frame, S: Silence frame

Criterion	Description	Real case	Detected case
Correct	Correct decisions for both (voiced and unvoiced)		
FEC (Front end clipping)	Clipping due to speech misclassified as noise in passing from noise ⇔ speech activity		
MSC (mid speech clipping)	Clipping due to speech being misclassifspeech regionied as noise during a.		
Over (Carry over)	Noise interpreted as speech in passing from speech activity to noise.		
NDS (noise detected as speech)	Noise interpreted as speech within silence region.		

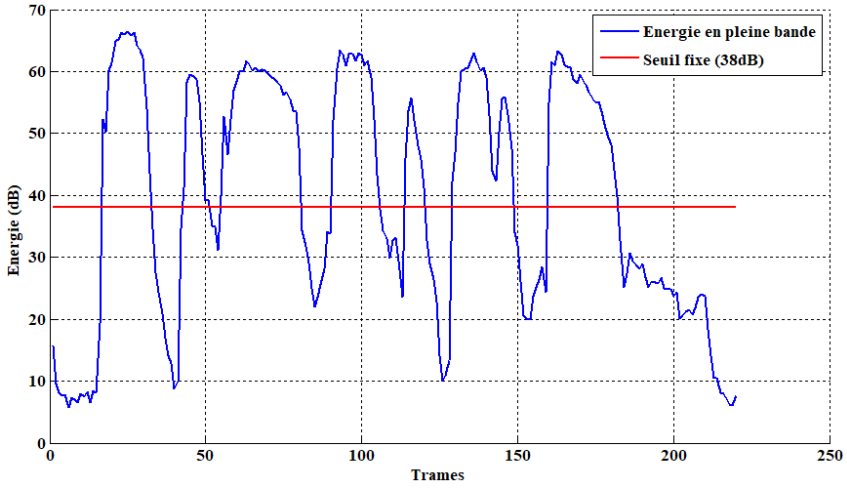
Over + NDS = FA (False Acceptance): silence frame detected as voiced

FEC + MSC = TR (True Rejection): true voiced frame rejected

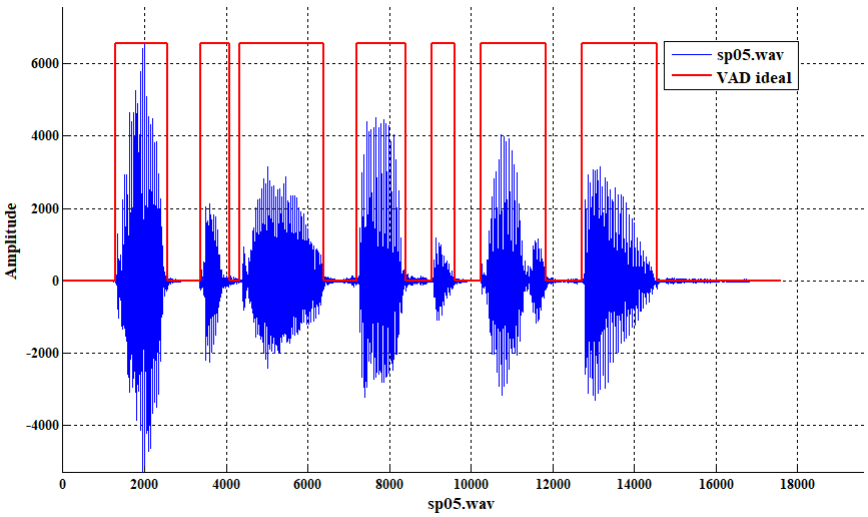
Figure 3.7: Critères de comparaison

Les critères ci-dessus expliqués ont été calculés en comparant la décision obtenue par notre approche avec la décision idéale pour chaque trame (utilisée référence de comparaison). Vu que la base de données NOIZEUS ne contient pas ces informations pour les 30 fichiers brutes, nous avons utilisé une technique de seuillage fixe, initialisé manuellement pour chaque phrase. Ce seuillage a été appliquée sur l'énergie en pleine bande de fréquence de chaque trame. Ensuite, afin d'éliminer les discontinuités indésirables, les résultats ont été lissés en inversant quelques trames singulières de silence situées en zone voisée et vis versa. A la fin, les trames déclarées comme silence sont mises à zéro et chaque phrase est réécoutée pour s'assurer de l'intelligibilité de l'allocution d'origine.

La Figure 3.8.a montre le résultat du VAD obtenu en appliquant la technique du seuillage fixe au fichier audio (sp05.wav La Figure 3.8.b met en évidence l'évolution de la décision VAD idéale en fonction du signal audio.



(a) Seuillage fixe de 38 dB



(b) VAD idéal obtenu

Figure 3.8: Seuillage fixe appliquée sur sp05.wav

3.4.3 Etude expérimentale

- **Calcul du facteur d'échelle T**

En utilisant l'équation (16), le facteur d'échelle T a été calculé dans un premier temps pour différents nombre de trames utilisées dans la phase initialisation N_0 de (6 à 20), et pour assurer différents taux d'acceptation dans l'intervalle [10^{-1} à 10^{-6}].

	N_0							
FA_{ex}	**6**	**8**	**10**	**12**	**14**	**16**	**18**	**20**
0,1	0,205010	0,153056	0,122105	0,101564	0,086938	0,075994	0,067497	0,060709
0,01	0,242273	0,180172	0,143398	0,119086	0,101821	0,088927	0,078931	0,070955
0,001	0,272700	0,202168	0,160601	0,133203	0,113788	0,099311	0,088101	0,079164
0,0001	0,299934	0,221747	0,175862	0,145699	0,124363	0,108475	0,096186	0,086397
0,00001	0,325288	0,239886	0,189958	0,157217	0,134096	0,116901	0,103613	0,093036
0,000001	0,349407	0,257063	0,203269	0,168073	0,143258	0,124824	0,110591	0,099270

Tableau 3.2: Valeurs du facteur d'échelle pour FA_{ex} et N_0 désirés

Toutes les valeurs trouvées dans le Tableau 3.2 ont été testées sur toute la base de données, pour extraire les meilleures combinaisons en se basant sur les critères *Correct*, *TR* et *FA*. D'après la simulation effectuée, nous avons retenu les valeurs préférentielles suivantes :

- N_0 variant de 6 à 12,
- FA_{ex} partielle égale à 0.1 ce qui donne une FA_{ex} globale de $0.1^p = 0.001$, sachant que le nombre des trames successives à lisser est $p = 3$.

- **Taux de fausse acceptation (False Acceptance Rate)**

Pour valider le principe du maintien du taux de fausse acceptation tel que défini lors de la conception de l'algorithme, nous avons concaténer plusieurs fois tous les fichiers qui représentent les 08 types de bruit : Passage des trains, Babillage, Voitures, Hall d'exposition, Restaurants, Rues, Aéroports et gares, dans le but de former un nombre de trames suffisant pour calculer la probabilité de fausse acceptation expérimentale. Le Tableau 3.3 illustre les résultats trouvés pour FA_{ex} partielle nominale 0.1 (FA_{ex} globale 0.001). Les résultats montrent qu'effectivement la moyenne FA_{ex} calculée $76x10^{-3}$est assez proche de la valeur du FA désiré 10^{-1} (nominale)

N_0	Nombre de Trames	FA_{ex} Partielle	FA_{ex} globale
6	8000	0,0753	0,000427
8	8000	0,0754	0,000429
10	8000	0,0763	0,000444
12	8000	0,0764	0,000446

Tableau 3.3: FA_{ex} partielles et FA_{ex} globales pour N_0 de 8 à 12

- **Seuil adaptatif**

Les paramètres d'ajustement du facteur d'échelle ont été fixés après d'intensives simulations ; rappelant que la condition de l'équation (17) doit être vérifiée chaque 200ms ($f_u = 20$). On note que la différence entre E_{min} et E_{max}, ΔE, est fixée à 20dB dans l'équation (18), le paramètre de niveau d'ajustement α est fixé à 0.1. Cette configuration est testée dans des conditions difficiles relatives à un environnement non stationnaire, ou la puissance du bruit de fond change très rapidement ; les Figures suivantes montrent la variation du seuil adaptatif par rapport à l'énergie pleine bande pour différents scénarios (Tableau 3.4) décrits par divers niveaux de puissance du bruit environnemental.

La simulation a été réalisée avec la concaténation de 16 signaux bruts de la base de données NOIZEUS. Le bruit de fond a été généré par le software Audacity [33] puis appliqué sur le signal concaténé pour différents niveaux de bruit de fond.

	Niveaux des SNR (dB)	Période pour chaque niveau SNR (secondes)	Durée du signal (secondes)
Scénario 1	[30 25 20 15 10 5 0]	6	42
Scénario 2	[0 5 10 15 20 25 30]	6	42
Scénario 3	[25 20 15 10 5 0 5 10 15 20 25]	3	33
Scénario 4	[0 5 10 15 20 25 20 15 10 5 0]	3	33

Tableau 3.4: Quatre scénarios testés pour le seuillage adaptatif

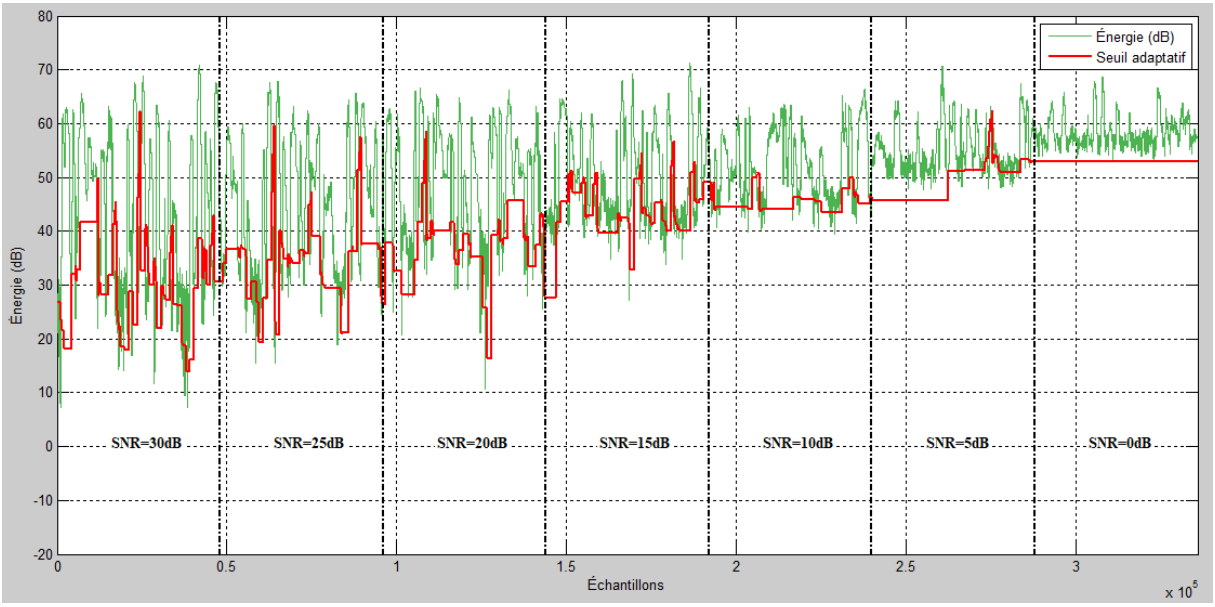


Figure 3.9: Variation du seuil adaptatif par rapport à l'énergie pour le scénario 1

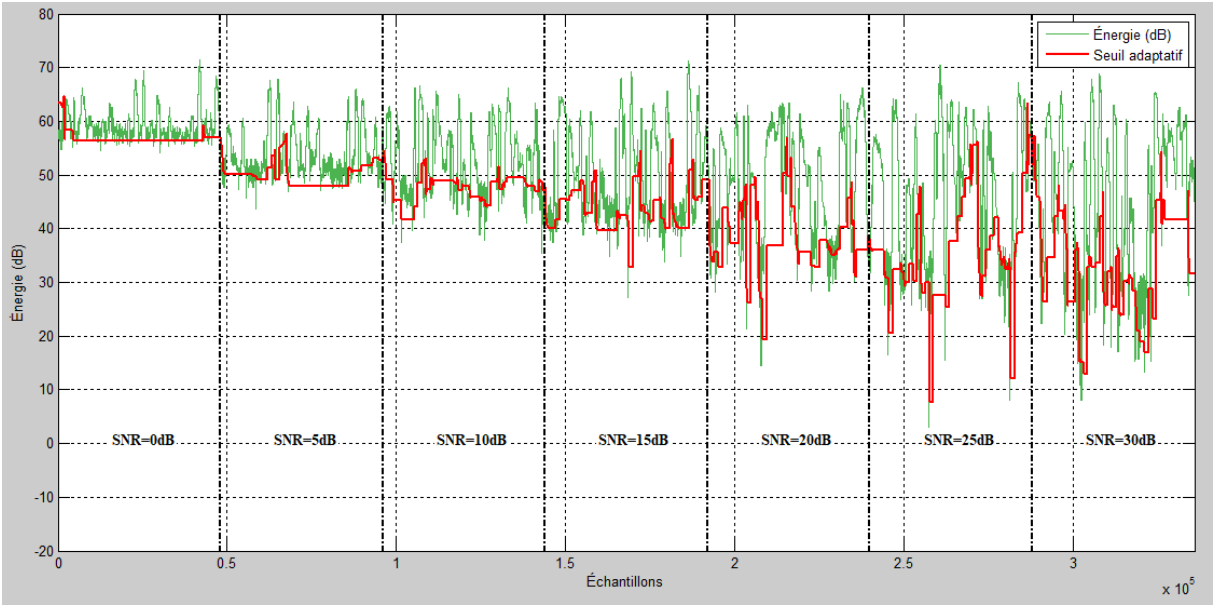


Figure 3.10: Variation du seuil adaptatif par rapport à l'énergie pour le scénario 2

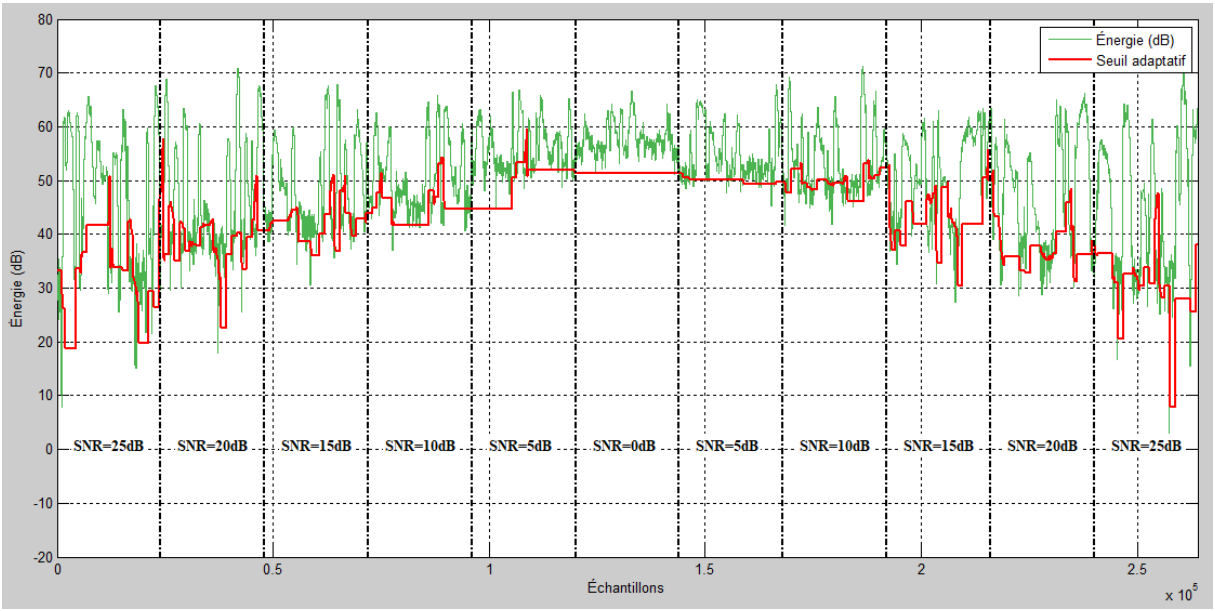


Figure 3.11: Variation du seuil adaptatif par rapport à l'énergie pour le scénario 3

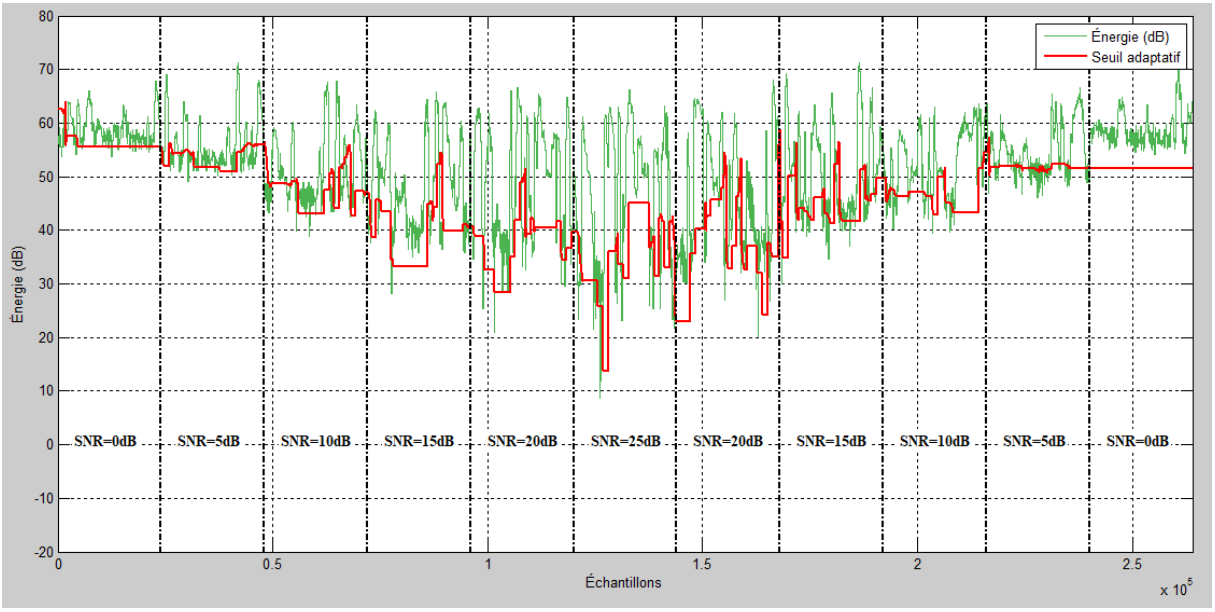


Figure 3.12: Variation du seuil adaptatif par rapport à l'énergie pour le scénario 4

3.4.4 Résultats dans un bruit stationnaire

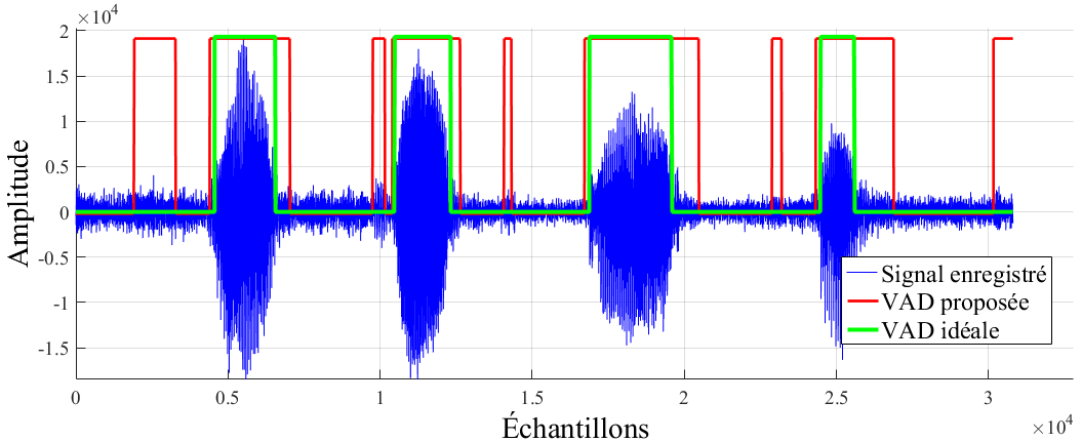
- **Signal réel enregistré :**

Un signal audio réaliste a été enregistré dans un environnement calme, pour expérimenter l'approche proposée dans un milieu stationnaire. Les décisions VAD, générées en appliquant à la fois l'algorithme proposé et le G.729, sont représentées sur les Figures 3.13.a et 3.13.b respectivement. Notant que les décisions VAD idéales ont été obtenues en étiquetant

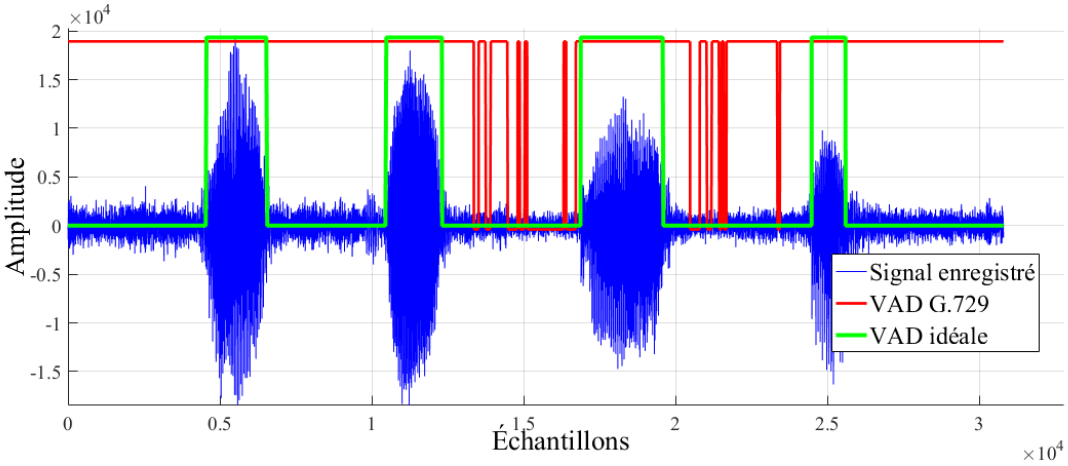
manuellement les trames de parole (Hand Labeling). Les paramètres d'évaluation obtenus, en utilisant le signal réel enregistré de la Figure 3.13 sont illustrés dans le Tableau 3.5.

Méthode	Correct	TR	FA	FEC	MSC	Over	NDS
G.729	35,84	0	64,15	0	0	1,03	63,11
Proposée	**78,96**	0	21,04	0	0	1,04	20

Tableau 3.5: La comparaison avec G.729 dans un environnement stationnaire



(a) Décision VAD de la méthode proposée



(b) Décisions VAD du standard G.729

Figure 3.13: Décisions VAD dans le cas « bruit stationnaire »

Comme le montre le Tableau 3.5, la valeur obtenue du critère *TR* (0%) indique qu'il n'y a pas de perte de trames vocales pour les deux méthodes. On observe également une augmentation intolérable de la Fausse Acceptation pour le G.729 (64,15%), ce qui conduit à une diminution

significative du paramètre Correct. Cependant, avec la méthode proposée on obtient une FA_{ex} très faible (21,04%), et par conséquent, une valeur relativement élevée du paramètre Correct (78,96%).

- **NOIZEUS**

Le Tableau 3.6 illustre les résultats obtenus pour l'ensemble de la base de données NOIZEUS, en considérant les paramètres opérationnels suivants, à savoir, $N_0 = 8$, $P = 3$ *et* $T = 0,13$ ce qui correspond à un FA nominal de $0,06x10^{-3}$. En outre, la comparaison de la méthode proposée avec le VAD G.729 pour différents types de dégradation à différents SNR, est présentée dans le Tableau 3.5. Notant que les meilleures performances pour chaque cas sont indiquées en gras pour les scores Correct, TR et FA_{ex}.

SNR (dB)	Méthode	Correct	TR	FA_{ex}	FEC	MSC	OVER	NDS
15	G.729	80,44	5.70	13.86	1.02	4.69	1.22	12.64
	Proposée	**85.06**	**2.73**	**12.20**	0.60	2.13	1.60	10.60
10	G.729	79.54	4.46	16	0.76	3.70	1.49	14.51
	Proposée	**83.70**	**3.81**	**12.49**	0.73	3.08	1.60	10.88
5	G.729	78.93	**3.69**	17.38	0.51	3.18	1.67	15.71
	Proposée	**82.80**	4.73	**12.46**	0.78	3.95	1.58	10.88
0	G.729	78.06	4.00	17.94	0.44	3.57	1.72	16.22
	Proposée	**80.92**	**6.11**	**12.96**	0.81	5.30	1.60	11.36
Moyenne	G.729	79.24	4.46	16.30	0.68	3.78	1.53	14.77
	Proposée	**83.12**	**4.35**	**12.53**	0.73	3.62	1.60	10.93

Tableau 3.6: La comparaison avec G.729 en utilisant NOIZEUS

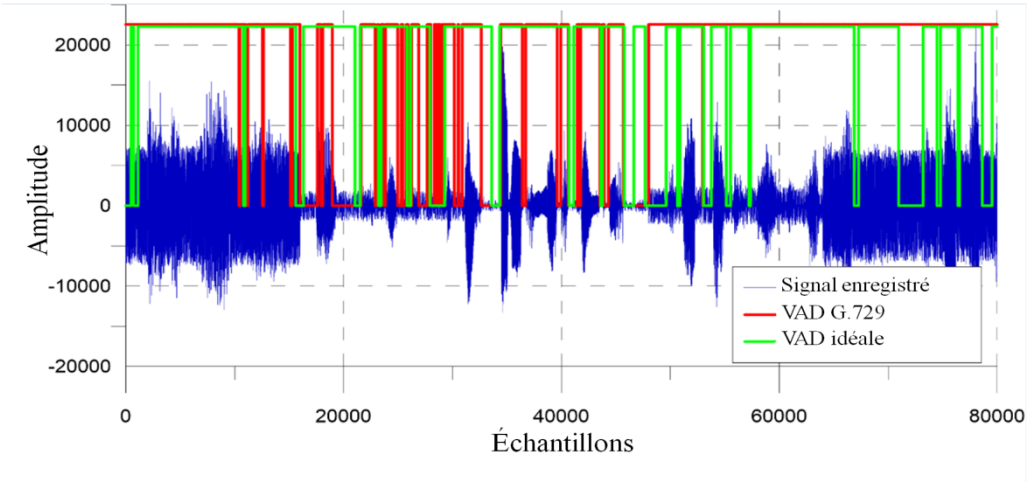
En analysant la moyenne des critères sur l'ensemble des données, on observe que le VAD proposé surpasse significativement la norme G.729-B, notamment en termes de paramètre Correct. En fait, une valeur élevée du paramètre Correct indique une performance de détection acceptable, tandis qu'un FA faible implique un bon taux de compression.

3.4.5 Résultats avec bruit de fond non-stationnaire

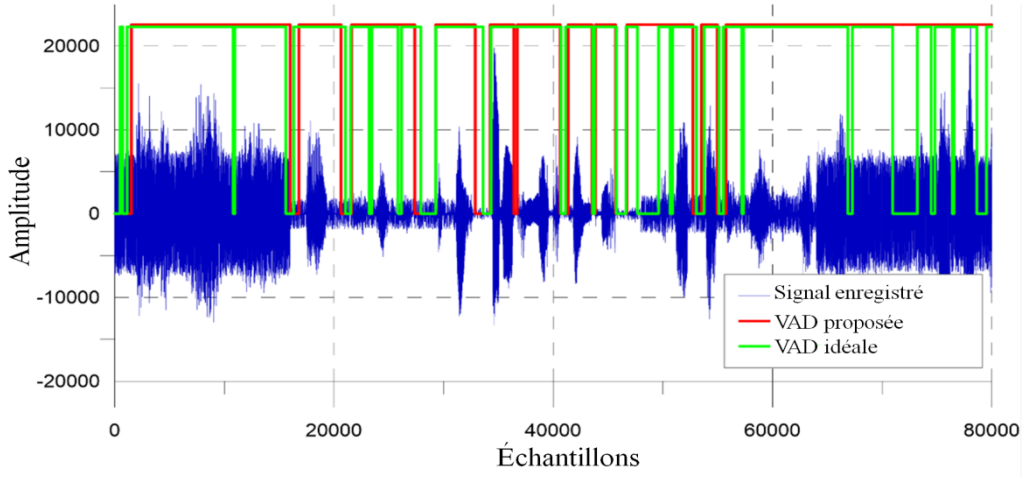
- **Signal réel enregistré :**

Pour analyser notre méthode VAD dans des environnements non stationnaires, quatre fichiers non corrompus ont été extraits de la base de données NOIZEUS et concaténés pour créer un fichier wav clair de 10s. Un bruit blanc synthétique est ensuite généré à partir de l'éditeur

audio Audacity pour l'ajouter au signal à différents SNR [-5dB, 10dB, 25dB, 10dB, -5dB]. Pour simuler des situations bruyantes non stationnaires avec différentes transitions de puissances, des signaux de bruit avec différents SNR sont ajoutés séquentiellement à la parole claire, chacun d'eux est appliqué pendant 2 secondes. Les Figures 3.14.a et 3.14.b illustrent les résultats des décisions engendrés par l'application à la fois de G.729B et de la méthode proposée respectivement. Les résultats des VAD obtenus sont résumés dans le Tableau 3.7 ci-dessous.



(a) Décisions VAD du standard G.729



(b) Décisions du VAD proposée

Figure 3.14: Décisions VAD dans le cas « bruit non-stationnaire »

Méthode	Correct	TR	FA	FEC	MSC	Over	NDS
G.729	75,3	13,1	11,6	1,3	11,8	2,7	8,9
Proposée	**84,6**	5,7	9,7	1,7	4	2,5	7,2

Tableau 3.7: La comparaison avec G.729 dans un environnement non-stationnaire

Les résultats obtenus montrent clairement les performances de la technique proposée dans un environnement non-stationnaire très bruité. En ce qui concerne le taux *Correct*, *TR* et *FA*, la technique proposée surpasse considérablement l'algorithme normalisé G.729.

- **Environnement non-stationnaire expérimental:**

Pour montrer la robustesse de l'approche proposée, nous l'avons expérimenté dans un environnement non stationnaire, en concaténant les fichiers audio suivants :
sp21.wav, sp29.wav, sp25.wav, 16.wav, sp10.wav, sp04.wav. Le but étant de former un signal global de 12 seconds qui sera mélangé à un bruit blanc généré par le logiciel Audacity pour créer un environnement expérimental avec des variations rapides des niveaux de puissance du bruit. Ainsi, quatre scénarios ont été créés et utilisés dans ce test (Tableau 3.8). Les résultats obtenus en appliquant la méthode proposée et le VAD du standard G.729 sont résumés dans le Tableau 3.9 pour chaque scénario.

	Niveaux des SNR (dB)	**Période pour chaque niveau SNR (secondes)**	**Période du signal (secondes)**
Scénario 1	[20 15 10 5 0 -5]	2	12
Scénario 2	[-5 0 5 10 15 20]	2	12
Scénario 3	[-5 0 5 10 15 10 5 0 -5]	1	9
Scénario 4	[15 10 5 0 -5 0 5 10 15]	1	9

Tableau 3.8: Quatre scénarios pour un environnement non-stationnaire

	Méthode	**Correct**	**TR**	**FA**	**FEC**	**MSC**	**Over**	**NDS**
Scénario 1	Proposée	**77,08**	**3,16**	19,74	0,33	2,83	4,16	15,58
	G729	76,83	4,49	18,66	0,33	4,16	3,83	14,83
Scénario 2	Proposée	**79,41**	**3,66**	16,91	0,5	3,16	3,91	13
	G729	69,41	26,74	3,82	3,41	23,33	1,16	2,66
Scénario 3	Proposée	**78,33**	**9,21**	12,44	0,77	8,44	3,22	9,22
	G729	62,22	31,77	5,99	2,66	29,11	1,22	4,77
Scénario 4	Proposée	**84,77**	**2,88**	12,33	0,55	2,33	3,33	9
	G729	81,22	5,55	13,21	0,55	5	3,44	9,77
Moyenne	Proposée	**79,9**	**4,73**	15,36	0,54	4,19	3,66	11,7
	G729	72,42	17,14	10,42	1,74	15,4	2,41	8,01

Tableau 3.9: Comparaison avec le G.729 dans un milieu non stationnaire expérimental

En observant les résultats obtenus dans le Tableau 3.9, la moyenne des scores obtenus (Correct, TR, FA) sont plutôt favorables à l'approche proposée comparativement aux standard G.729.

Conclusion

Dans ce chapitre, nous avons présenté l'état de l'art des principales méthodes VAD basées sur les décisions statistiques. De cette étude, il en découle que les distributions de Laplace et de Gauss sont les plus indiquées comme modèles statistiques pour décrire la voix active et le bruit de fond au niveau d'un signal audio.

En termes de contribution, nous avons également présenté un schéma de détection de l'activité vocale peu complexe qui exploite l'énergie pleine bande du signal audio. L'algorithme proposé dans ce cadre fonctionne d'une manière séquentielle en générant, pour chaque trame audio, une décision binaire relative à la présence ou à l'absence de la voix active au niveau de la trame sous test. Afin de faire face à la variabilité du niveau de puissance induite par la non stationnarité des divers environnements bruités, nous avons mis en œuvre une stratégie de détection basée sur le maintien du taux de fausse acceptation. Par analogie à la détection radar, ce taux est équivalent à la fausse alarme et correspond à la probabilité pour que des trames de silence soient *faussement* classées comme voix active. Ce principe permet au détecteur de s'adapter dynamiquement aux variations de l'environnement ambiant et d'ajuster le seuil de détection relativement à une mesure locale de l'énergie des trames précédent celle en cours de traitement. A l'instar de plusieurs VAD normalisés, le traitement local dit « trame par trame » (frame by frame processing) génère un phénomène de hachage au niveau des décisions VAD, ce qui nous a motivés à développer notre propre procédure de lissage qui s'articule sur le résultat de la trame sous test, mais également sur l'analyse d'un nombre réduit de décisions partielles précédemment générées.

Enfin, le VAD proposée a été comparée à celui du standard G.729 en utilisant la base de données audio NOIZEUS. Pour compléter notre étude comparative et afin de valider la robustesse de l'algorithme proposé, nous avons effectué une série de tests expérimentaux dans des situations assez réalistes, en créant divers scénarios dans des environnements bruités stationnaires et non stationnaires.

Chapitre 4:
Implémentation du VAD proposé

Résumé

Dans ce chapitre, nous présentons l'état de l'art dans le domaine de l'implémentation des méthodes VAD sur divers types de matériel (DSP, FPGA et prothèses auditives), par la suite, nous décrivons l'architecture générale de la carte STM32F7, utilisée en tant que système cible. Les parties hardwares et softwares développées dans l'implémentation de notre algorithme VAD seront examinés en détails tout en expliquant l'ensemble des tests exécutés en mode temps réel, via plusieurs enregistrements audio effectués dans divers environnements bruités.

4.1 Introduction

La méthode VAD proposée est implémentée sur le Microcontrôleur STM32F746NG et testée par le biais d'expérimentations réelles dans divers environnements. Dans ce chapitre, nous donnons les principales caractéristiques matérielles concernées par l'implémentation (CPU, Codec, microphones, mémoires, SAI, DMA, …etc.), ainsi que les environnements de développements intégrés qui peuvent être utilisés (STM32CubeMX, IAR Embedded Workbench,…etc.). Ensuite, nous expliquons les détails de la solution d'implémentation de l'architecture proposée, puis nous procédons à l'analyse de sa robustesse en se focalisant sur les contraintes « temps réel », via l'outil logiciel STM-STUDIO.

4.2 Implémentation hardware des VAD

- **Etat de l'art**

Dans [34], les auteurs ont développé une architecture de détection d'activité vocale en combinant la méthode de soustraction spectrale et de l'énergie à court terme. L'approche proposée a été implémenté sur circuit FPGA (Field Programmable Gate Array) en utilisant Xilinx System Generator (XSG) et la carte Digilent Nexys-4, afin de réaliser le rehaussement de la parole noyée dans un bruit additif non-stationnaire. En terme de complexité d'implémentation, les résultats obtenus dans cette étude font ressortir le pourcentage des ressources occupées par rapport aux entités matérielles disponibles au niveau du circuit FPGA cible, en particulier le nombre de bascules (Flip Flpos), le nombre LUT (Look up Tables) et le nombre de slices. Pour un circuit FPGA de moyenne densité cadencé à une fréquence de 67.09 MHz, les taux d'occupation nécessaires à l'implémentation de cet algorithme ont été [34] comme suit: Slice (16.3%), Flip Flops (7.6%), LUTs (13.4%). Ces taux d'occupation, relativement faibles, prouvent que le VAD implémenté possède une structure algorithmique peu complexe.

Dans [35], une version modifiée de l'algorithme Nortel VAD-CNG a été implémentée sur une carte DSP (TMS320C5402DSK). Les résultats expérimentaux ont montré que la complexité du module VAD a été réduite de 86%. En termes de vitesse d'exécution, exprimée en MIPS (Million of Instructions Per Second), il a été montré que le bloc VAD a été optimisé en le réduisant la contrainte liée à la vitesse de calcul de 26.65 MIPS à 3.70 MIPS [35].

Moyennant un équipement plus spécifique, N. Lezzoum et *al* ont proposé un algorithme VAD qui utilise la comparaison entre l'énergie dans la région de fréquence contenant des informations vocales et celles contenant uniquement du bruit [36]. La décision du VAD est prise à l'aide de deux règles de comparaison de seuils, calculés à partir des caractéristiques normalisées et d'un schéma de basculement déclenché après un nombre donné d'observations. L'approche proposée a été implémentée sur la plateforme ARP (Auditory Research Platform) [37]. Le nombre d'instructions par trame audio traitée obtenu a été de l'ordre de 890 [36], ce qui équivaut à un taux de 87% de la taille totale de la RAM programme. La RAM de données utilisée par ce VAD est de l'ordre de 8% de l'ensemble de la RAM data totale, tandis que le coefficient RAM utilisé est de 240 (23% du coefficient RAM).

4.3 Microcontrôleur STM32F746NGH6:

4.3.1 Introduction :

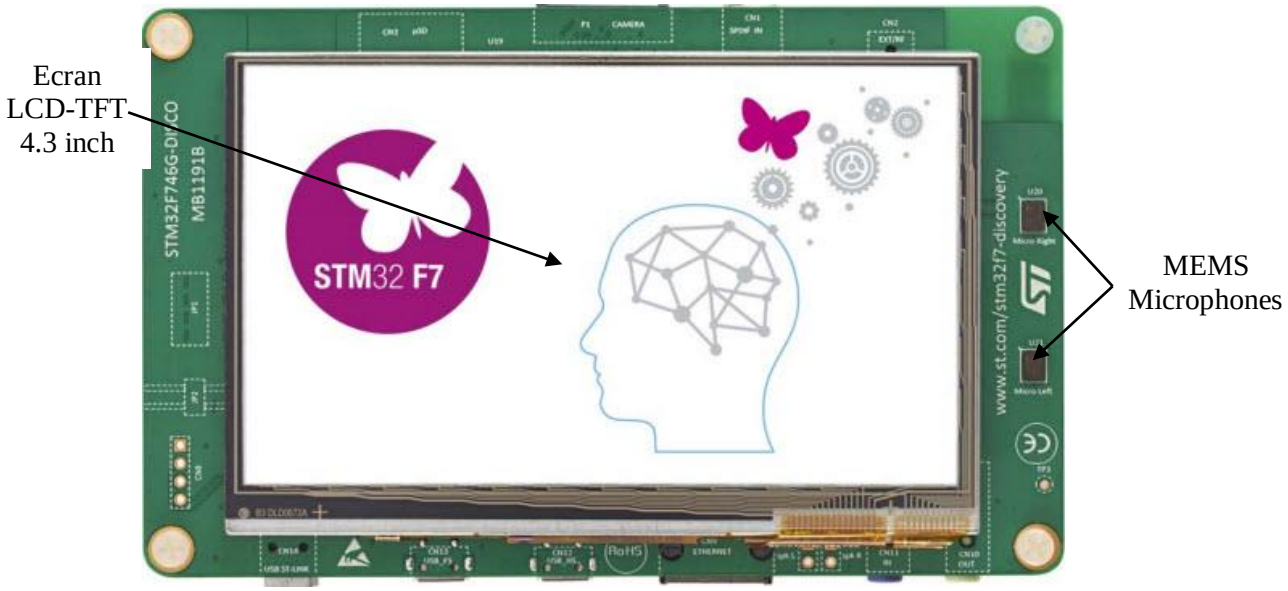
La carte STM32F746G-DISCOVERY est une plate-forme complète de développement pour le microcontrôleur STM32F746NGH6 basé sur le cœur STMicroelectronics ARM® Cortex®-M7 [38]. Ce microcontrôleur autorise une fréquence d'horloge allant jusqu'à 216 MHz, possède quatre interfaces bus du type I^2C (Inter Integrated Circuit Bus), six SPI (Serial Peripheral Interface) avec trois périphériques I2S (Inter Integrated Sound) simplex multiplexés, SDMMC (Secure Digital and Multi Media Card), quatre interfaces sérielles USART (Universal Synchronous/Asynchronous Receiver Transmitters), quatre interfaces UART, deux CAN (Controller area network), trois ADC (Analog-to-Digital Converters) 12 bits, deux DAC (Digital-to-Analog Converters) 12 bits, deux interfaces SAI (serial audio interface), une mémoire interne SRAM (Static Random Access Memory) de 320 Kb et une mémoire Flash de 1 Mo.

Cette carte offre tout ce dont les utilisateurs ont besoin pour développer facilement des applications. La gamme complète de fonctionnalités matérielles sur la carte aide les utilisateurs à évaluer presque tous les périphériques intégrés au système; à savoir: SAI Audio, DAC stéréo avec entrée et sortie jack audio, microphones numériques MEMS (Micro Electro Mechanical System), écran multitouche capacitif LCD-TFT,…etc.

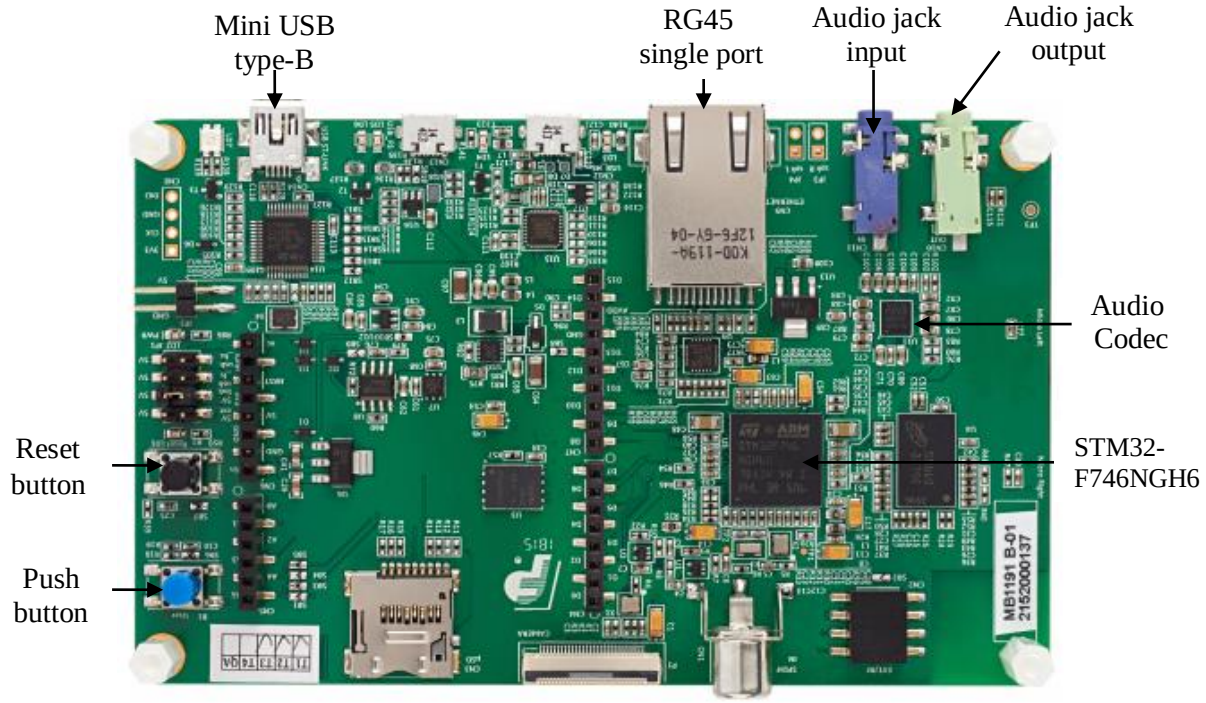
La carte STM32F746G-DISCOVERY est livrée avec le logiciel complet STM32 HAL (Hardware Abstraction Layer), ainsi que divers exemples fournis, en plus d'un accès direct à ARM® mbed ™ ressources en ligne via *http://mbed.org*

Le système STM32F746NGH6 se présente sous l'aspect matériel suivant :

La Figure 4.1.a et 4.1.b montre l'interface de la carte vue de haut et vue de bas respectivement.



(a) Vue de haut de STM32F746G-DISCO



(b) Vue de bas de STM32F746G-DISCO

Figure 4.1: Carte STM32F746G-DISCO

4.3.2 Description matérielle:

Le bloc diagramme de la carte est montré dans la Figure 4.2.

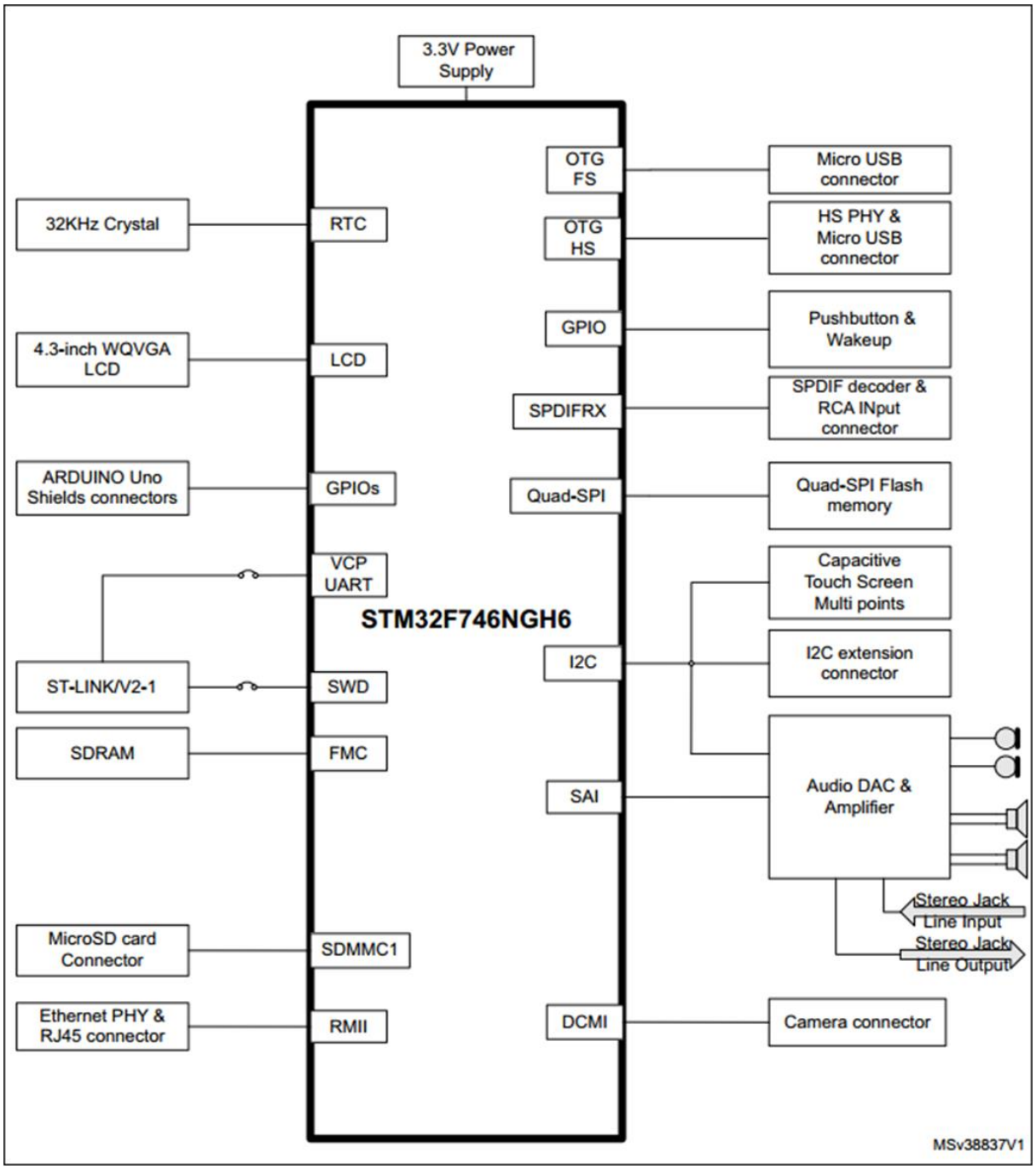


Figure 4.2: Bloc diagramme de STM32F746NGH6

Les caractéristiques matérielles principales utilisées dans notre implémentation sont expliquées ci-dessous :

- **Audio**

Un codec audio WM8994ECS/R de CIRRUS avec 4 DAC et 2 ADC est connecté à l'interface SAI du STM32F746NGH6, il communique avec STM32F746NGH6 via le bus I2C partagé avec le module caméra et le connecteur d'extension I2C.

- L'entrée de ligne analogique est connectée au CAN du WM8994ECS/R via la prise CN11.
- La sortie de ligne analogique est connectée au DAC du WM8994ECS/R via la prise audio CN10.
- Deux haut-parleurs externes peuvent être connectés au WM8994ECS/R via JP3 pour le haut-parleur droit et JP4 pour le haut-parleur gauche.
- Deux microphones numériques (microphone ST MEMs) MP34DT01TR se trouvent sur la carte STM32F746G-DISCO. Ils sont connectés aux microphones numériques d'entrée du WM8994ECS/R.
- Un connecteur coaxial CN1 est implémenté sur le STM32F746G-DISCO pour recevoir des données audio externes compatibles avec la spécification SPDIF.

- **Analog-to-Digital Converters (ADCs)**

Trois convertisseurs analogique-numérique 12 bits sont intégrés et chaque ADC partage jusqu'à 16 canaux externes, effectuant des conversions en mode single-shot ou scan. En mode de balayage, la conversion automatique est effectuée sur un groupe sélectionné d'entrées analogiques.

Des fonctions logiques supplémentaires intégrées dans l'interface ADC permettent:

- Échantillonnage et maintien simultanés
- Échantillon et prise entrelacés

L'ADC peut être servi par le contrôleur DMA (Direct Memory Access). Une fonction de surveillance analogique permet une surveillance très précise de la tension convertie d'un ou de plusieurs canaux sélectionnés. Une interruption est générée lorsque la tension convertie est en dehors des seuils programmés.

- **Digital-to-Analog Converter (DAC)**

Les deux canaux DAC 12 bits peuvent être utilisés pour convertir deux signaux numériques en deux sorties de signaux de tension analogiques.

Cette double interface numérique prend en charge les fonctionnalités suivantes:

- Deux convertisseurs DAC: un pour chaque canal de sortie
- Alignement des données à gauche ou à droite en mode 12 bits

- Capacité de mise à jour synchronisée
- Génération de signaux de bruit
- Génération d'ondes triangulaires
- Conversions simultanées ou indépendantes de deux canaux DAC
- Capacité DMA pour chaque canal
- Déclencheurs externes de conversion

Huit entrées de déclenchement DAC sont utilisées dans l'appareil. Les canaux DAC sont déclenchés via les sorties de mise à jour du Timer qui sont également connectées à différents flux DMA.

- **Ecran LCD-TFT**

Le dispositif d'affichage couleur (480x272 LCD-TFT) avec écran tactile capacitif est connecté à l'interface LCD RGB du STM32F746NGH6.

- **SAI (Serial Audio Interface)**

Les appareils intègrent deux interfaces audio série. L'interface audio série est basée sur deux sous-blocs audio indépendants qui peuvent fonctionner comme émetteur ou récepteur avec leur mémoire FIFO. De nombreux protocoles audio sont pris en charge par chaque bloc: Normes I2S, justifiés LSB ou MSB, sortie PCM / DSP, TDM, AC'97 et SPDIF, prenant en charge des fréquences d'échantillonnage audio de 8 kHz à 192 kHz. Les deux sous-blocs peuvent être configurés en mode maître ou en mode esclave.

En mode maître, l'horloge maître peut être émise vers le DAC / CODEC externe à 256 fois la fréquence d'échantillonnage.

Les deux sous-blocs peuvent être configurés en mode synchrone lorsqu'un mode duplex intégral est requis. SAI1 et SAI2 peuvent être servis par le contrôleur DMA.

- **DMA (Direct Memory Access)**

La carte dispose de deux DMA à double port à usage général (DMA1 et DMA2) avec 8 flux chacun. Ils sont capables de gérer les transferts de mémoire à mémoire, de périphérique à mémoire et de mémoire à périphérique. Ils disposent de FIFO dédiés pour les périphériques APB (Advanced Peripheral Bus) / AHB (Advanced High-performance Bus), prennent en charge le transfert en rafale et sont conçus pour fournir la bande passante périphérique maximale (AHB / APB).

Les deux contrôleurs DMA prennent en charge la gestion de la mémoire tampon circulaire, de sorte qu'aucun code spécifique n'est nécessaire lorsque le contrôleur atteint la fin de la mémoire tampon. Les deux contrôleurs DMA ont également une fonction de double tampon, qui automatise l'utilisation et la commutation de deux tampons mémoire sans nécessiter de code spécial.

Chaque flux est connecté à des requêtes DMA matérielles dédiées, avec prise en charge du déclencheur logiciel sur chaque flux. La configuration est effectuée par logiciel et les tailles de transfert entre la source et la destination sont indépendantes.

Le DMA peut être utilisé avec les principaux périphériques:

- SPI and I2S
- I^2C
- USART
- GPIOs (General-purpose for Input/Outputs)
- TIMx (Timers) basiques ou avec contrôle avancé
- DAC et ADC,
- SDMMC
- SAI

- **GPIOs**

Chacune des broches GPIO peut être configurée par logiciel comme sortie, comme entrée ou comme fonction alternative périphérique. La plupart des broches GPIO sont partagées avec des fonctions alternatives numériques ou analogiques. Tous les GPIO sont compatibles avec un courant élevé et disposent d'une sélection de vitesse pour mieux gérer le bruit interne, la consommation d'énergie et les émissions électromagnétiques.

La configuration des entrés / sorties peut être verrouillée si nécessaire en suivant une séquence spécifique afin d'éviter une écriture parasite dans les registres. La gestion rapide des entrés / sorties permettant un basculement maximal jusqu'à 108 MHz.

- **System Trace Macrocell (STM)**

L'ARM Embedded Trace Macrocell offre une meilleure visibilité des instructions et du flux de données à l'intérieur du cœur du processeur en diffusant des données compressées à un débit très élevé depuis le STM32F74xxx via un petit nombre de broches ETM vers un périphérique TPA (Trace Port Analyzer) externe. Le TPA est connecté à un ordinateur hôte

via l'interface USB, Ethernet ou tout autre canal haut débit. Les instructions en temps réel et l'activité de flux de données peuvent être enregistrées, puis formatées pour être affichées sur l'ordinateur hôte qui exécute le logiciel de débogage.

- **Codec WM8994**

Le WM8994 [39] est un codec audio de haute qualité conçu pour s'interfacer avec une large gamme de processeurs et de composants analogiques. Un haut niveau d'intégration de signaux mixtes le rend idéal pour les applications portables telles que les téléphones mobiles. L'architecture interne, entièrement différentielle, et les filtres de bruit On-Chip RF garantissent un très haut degré d'immunité au bruit.

Les ADC et DAC, utilisent un sur échantillonnage sur 24 bits pour offrir des performances optimales. Un agencement d'horloge flexible prend en charge des fréquences d'échantillonnage mixtes, tandis que des circuits FLL (Frequency-Locked Loop) intégrées offrent une flexibilité supplémentaire. Un filtre passe-haut est disponible dans toutes les voies ADC et MIC numériques pour supprimer les composantes continues (DC) et les bruits de basse fréquence.

Le WM8994 possède des interfaces audio numériques très flexibles, prenant en charge un certain nombre de protocoles, y compris I2S, DSP, MSB-first, et peut fonctionner en mode maître ou esclave. Le fonctionnement au format PCM est pris en charge en mode DSP. Le multiplexage par répartition de temps TDM (Time Division Multiplexing) est disponible pour permettre à plusieurs appareils de diffuser des données simultanément sur le même bus.

L'horloge système (SYSCLK) fournit une horloge pour les ADC, les DAC, le cœur DSP, l'interface audio numérique et d'autres circuits. SYSCLK peut être dérivée directement de l'une des broches MCLK1 ou MCLK2 ou via une de deux FLL intégrés, Les fréquences MCLK typiques des systèmes portables et les fréquences d'échantillonnage de 8 kHz à 96 kHz sont toutes prises en charge.

La fonctionnalité GPIO polyvalente est fournie, avec prise en charge des entrées de détection de bouton / accessoire, ou pour l'horloge, l'état du système ou la sortie de niveau logique programmable pour le contrôle de circuits externes supplémentaires.

4.3.3 Environnements de développement:

Les microcontrôleurs et microprocesseurs ont toujours exigé des assembleurs, des compilateurs et des éditeurs de liens ainsi que des logiciels de débogage et de programmation. Les développeurs de logiciels doivent disposer d'une large gamme d'environnements de développement intégrés (IDE Integrated Development Environment) capables de conFigurer et d'initialiser le MCU (Microcontroller Unit) ou le MPU (Memory Protection Unit) ainsi que de surveiller son comportement au moment de l'exécution.

Ces outils sont disponibles à la fois auprès de ST et auprès d'un large choix de fournisseurs tiers. La plupart des outils sont disponibles en téléchargement gratuit dans les pages suivantes [40]. Actuellement, ST fournit 28 environnements de développement, listés dans [40], ainsi, plusieurs softwares d'analyse et de supervision en temps réel sont disponibles [41], nous montrons ci -dessous l'EDI originale fournit par ST, ainsi que les softwares utilisés dans notre implémentation.

- **STM32CubeMX**

STM32CubeMX [42] est une initiative originale de STMicroelectronics visant à faciliter la le développement en réduisant le temps et les coûts de développement. C'est un outil graphique qui permet une configuration très simple des microcontrôleurs et microprocesseurs STM32, ainsi que la génération du code C d'initialisation correspondant pour le cœur Arm® Cortex, à travers un processus étape par étape.

La Figure 4.3 montre l'interface graphique du STM32CubeMX.

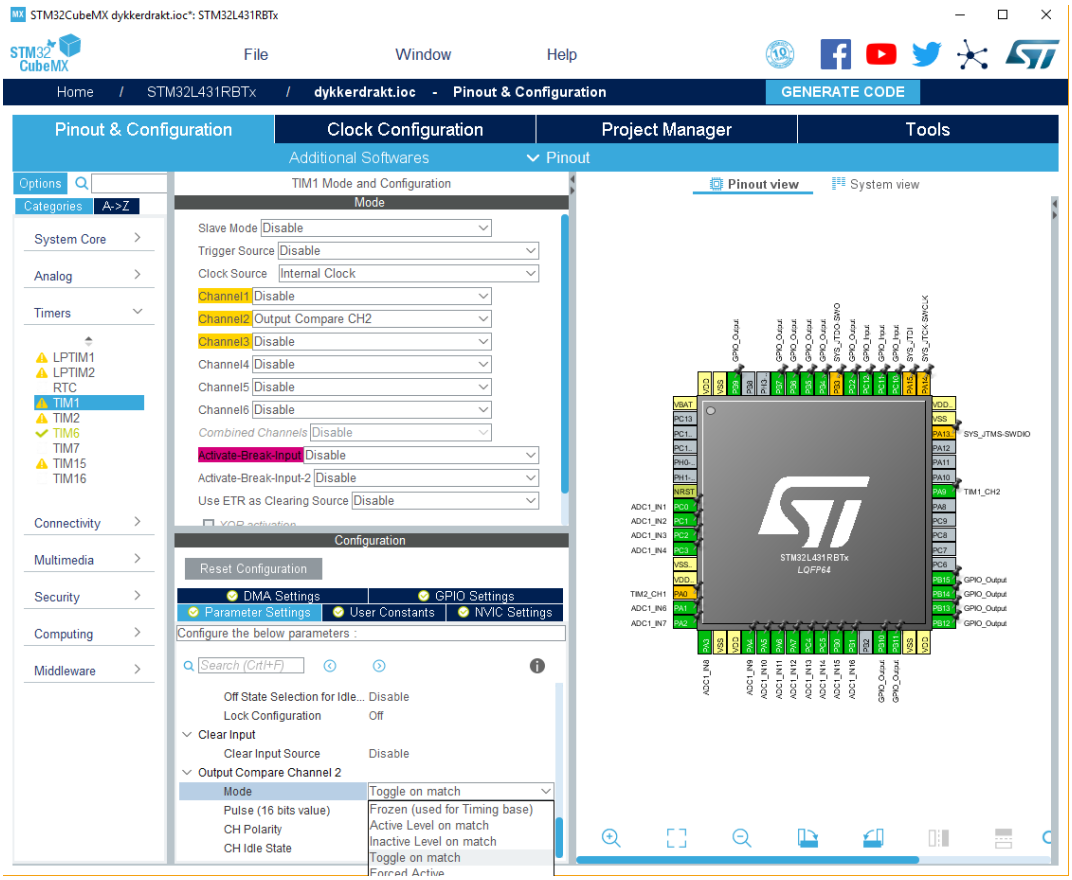


Figure 4.3: Interface graphique de STM32CubeMX

- **IAR Workbench**

IAR Embedded Workbench [43], est un environnement de développement intégré, et le compilateur IAR C/C ++ inclus génèrent un code performant et compact pour des applications Arm®.

C-SPY Debugger, est un débogueur entièrement intégré avec analyse des performances, visualisation de l'alimentation et prise de RTOS (Real Time Operating System).

Les fichiers de configuration, les exemples de code et les modèles de projet sont inclus.

Cet EDI permet de créer un code plus petit, plus rapide et plus intelligent avec une fonctionnalité de débogage complète (analyse statique et d'exécution disponible, débogage de l'alimentation pour une consommation d'énergie minimisée).

L'interface graphique du logiciel est montrée dans la Figure 4.4.

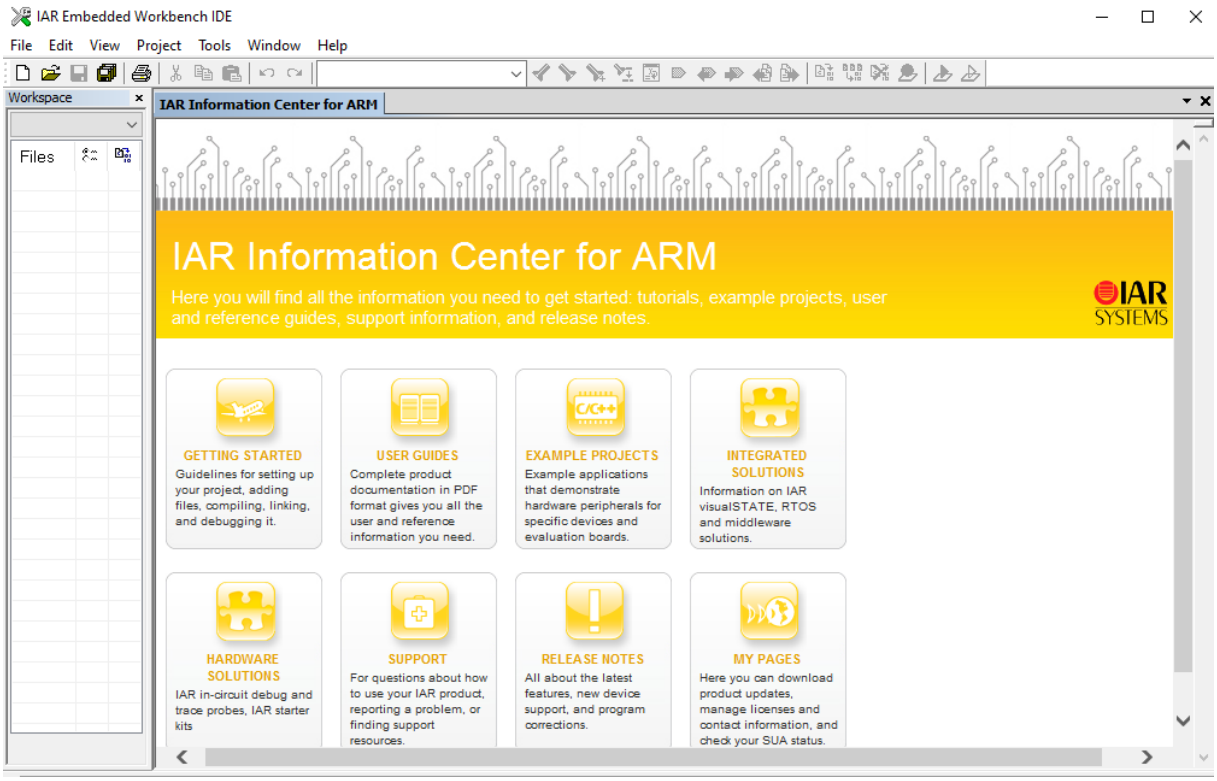


Figure 4.4: L'interface graphique d'IAR Workbench

- **STM-Studio**

STM Studio [44] permet de déboguer et de diagnostiquer les applications STM32 pendant leur exécution en lisant et en affichant leurs variables en temps réel.

Exécuté sur un PC, STM Studio s'interface avec les microcontrôleurs STM32 via les outils de développement standard ST-LINK.

STM Studio est un outil non intrusif, préservant le comportement en temps réel des applications. Il complète parfaitement les outils de débogage traditionnels pour affiner les applications. Il convient parfaitement aux applications de débogage qui ne peuvent pas être arrêtées, telles que les applications de commande de moteur.

Différentes vues graphiques sont disponibles pour répondre aux besoins de débogage et de diagnostic ou pour démontrer le comportement de l'application.

La Figure 4.5 montre l'interface graphique du software.

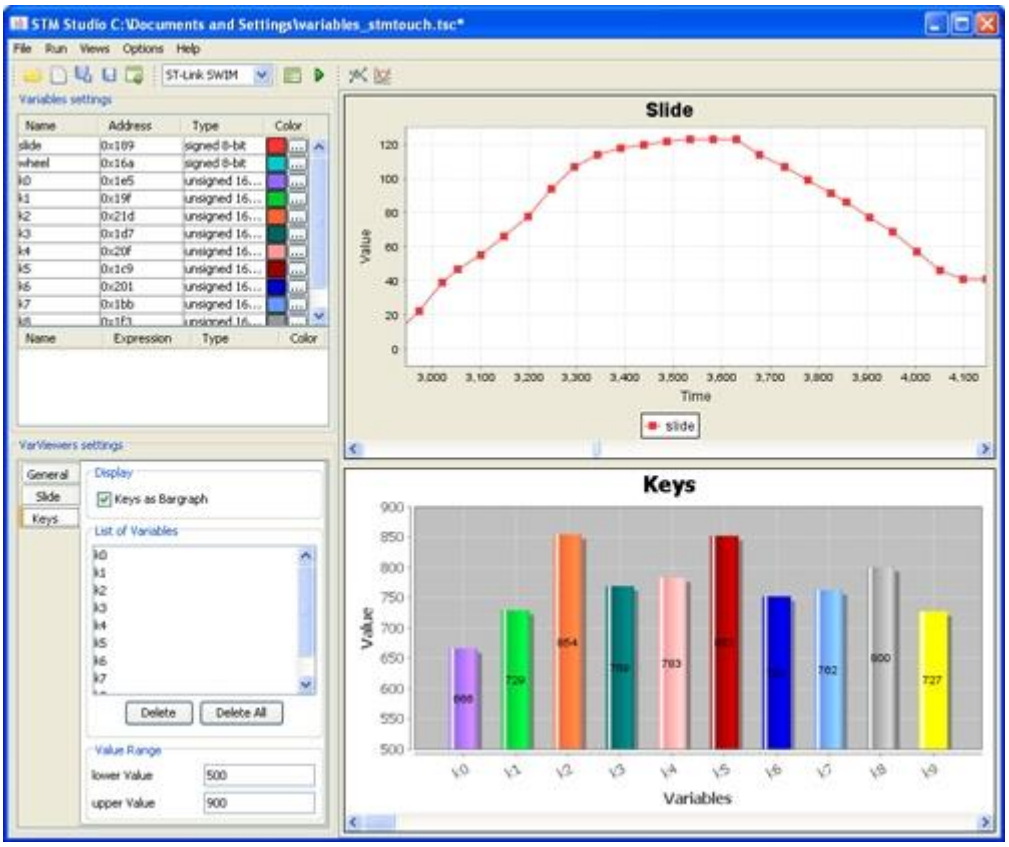


Figure 4.5: Interface graphique de STM-STUDIO

4.4 Implémentation du VAD proposée sur STM32F746

L'approche VAD proposée, décrite dans le chapitre 03, est implémentée dans la carte STM32F746. Son noyau Cortex®-M7 comprend une seule unité en virgule flottante prenant en charge toutes les instructions et types de données ARM® au format simple précision. La carte de développement incorpore des mémoires embarquées à haute vitesse avec une mémoire Flash jusqu'à 1 Mo et une vaste gamme de périphériques d'entrée / sortie. La Figure 4.6 illustre le fonctionnement du système en temps réel. Il se compose de trois parties fondamentales; une source audio (du microphone), une carte MCU et le haut-parleur. Le signal vocal analogique de la source audio passe par le convertisseur analogique-numérique (CAN) du codec WM8994ECS/R pour être échantillonné à 8 kHz et numérisé en données PCM linéaires (Pulse Code Modulation) 16 bits. Le signal numérique obtenu sera traité trame par trame via l'algorithme VAD proposé en temps réel. Enfin, les données numériques de sortie sont converties en un signal analogique via le convertisseur (CNA) et passées par le haut-parleur, ce qui permet une évaluation subjective de la méthode VAD étudiée.

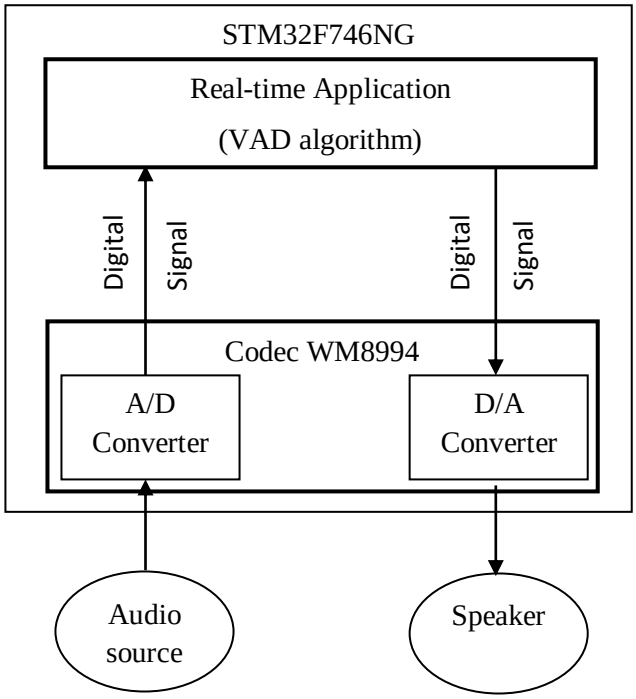


Figure 4.6: Architecture général du système VAD en temps réel

4.4.1 Conception de l'implémentation matérielle

Les pilotes BSP (Board Support Package) de la carte sont fournis pour configurer tout le matériel requis pour l'application audio (Codec, SAI, DMA, …etc.). Les Tableaux 4.1 et 4.2 résument la configuration principale du codec et des ressources matérielles de l'application audio, respectivement.

Périphérique	Registre	Valeur	Remarque
Codec WM8994ECS/R	INPUT_DEVICE_INPUT_LINE_1	0x0300	input device
	OUTPUT_DEVICE_HEADPHONE	0x0002	output device
	Niveau de volume	0x64	set to MAX
	I2S_AUDIOFREQ_8K	8000	Audio Frequency IN
	AUDIO_BLOCK_SIZE	0xA0	160 bytes for Half Block, 320 bytes for FULL Block

Tableau 4.1: Configuration du CODEC WM8994ECS/R

Périphériques	Registre	Valeur	Remarque
Serial Audio Interface (SAI)	Protocol	0x00000000U	SAI_FREE_PROTOCOL
	AUDIO_IN_SAIx	(SAI2_BASE + 0x024U)	SAI2_Block_B, conFigured as slave Rx mode synchronous from SAI2_block_A
	AUDIO_OUT_SAIx	SAI2_BASE + 0x004U)	SAI2_Block_A, conFigured as Master Rx Mode
DMA	AUDIO_IN_SAIx_DMAx_STREAM	(DMA2_BASE + 0x0B8U)	DMA2_Stream7
	AUDIO_IN_SAIx_DMAx_CHANNEL	0x00000000U	DMA_CHANNEL_0
	AUDIO_OUT_SAIx_DMAx_STREAM	(DMA2_BASE + 0x070U)	DMA2_Stream4
	AUDIO_OUT_SAIx_DMAx_CHANNEL	0x06000000U	DMA_CHANNEL_3

Tableau 4.2: Ressources matérielles de l'application AUDIO

La Figure 4.7 illustre le principe de fonctionnement de l'architecture matérielle. Les données audio analogiques sont échantillonnées en utilisant le convertisseur (CAN) du codec puis transférées vers le composant DMA (accès direct à la mémoire) par l'interface audio série (SAI). Les deux blocs A et B sont sélectionnés et configurés pour effectuer cette tâche. Comme la montre la Figure 4.8, toutes les 10 ms, nous obtenons un bloc complet rempli, qui représente une trame des données enregistrées en ligne. Ensuite, notre schéma VAD est appliqué pour obtenir, pour chaque trame, la décision finale (voisée ou non voisée).

Le code VAD, écrit en langage C dans le MCU, est compilé dans l'environnement de développement intégré IAR Embedded Workbensh. Il est ensuite téléchargé sur la carte via le périphérique USB à l'aide du compilateur IDE ou du logiciel STM intégré. L'horloge système utilisée est de 49,142 MHz.

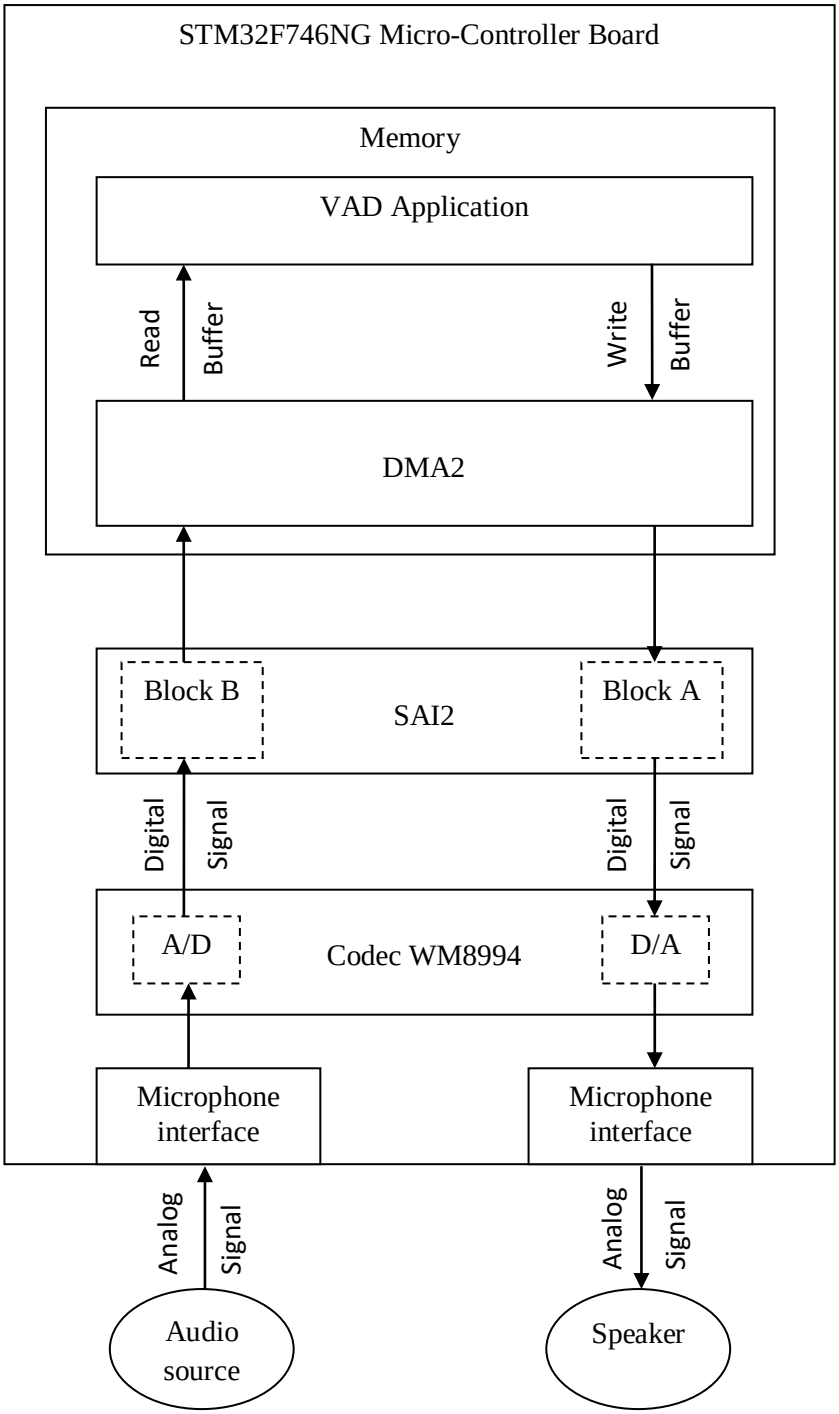


Figure 4.7: Flux de données du système VAD Temps Réel

4.4.2 Organisation des données en mémoire

Dans ce qui suit, nous décrivons d'abord le Mapping Mémoire requis pour stocker et traiter les trames audio entrantes. Nous présentons également un aperçu des principales variables et constantes utilisées par l'algorithme VAD proposé. Le Tableau 4.3 résume la signification, la taille, le type et les adresses des ressources mémoire utilisées. La structure de la mémoire de

données pour un bloc complet, correspondant à la trame audio entrante en cours, est représentée sur la Figure 4.8. Rappelons que la carte permet de gérer deux canaux d'entrée audio (gauche et droit) pour les opérations en mode stéréo. L'acquisition audio est effectuée en utilisant une fréquence d'échantillonnage de 8 kHz et un format de données PCM 16 bits. Par conséquent, 320 octets doivent être stockés et seulement 160 sont traités pour chaque trame de 10 ms en sélectionnant un seul canal dans un mode audio mono, tandis que les données du deuxième canal sont ignorées.

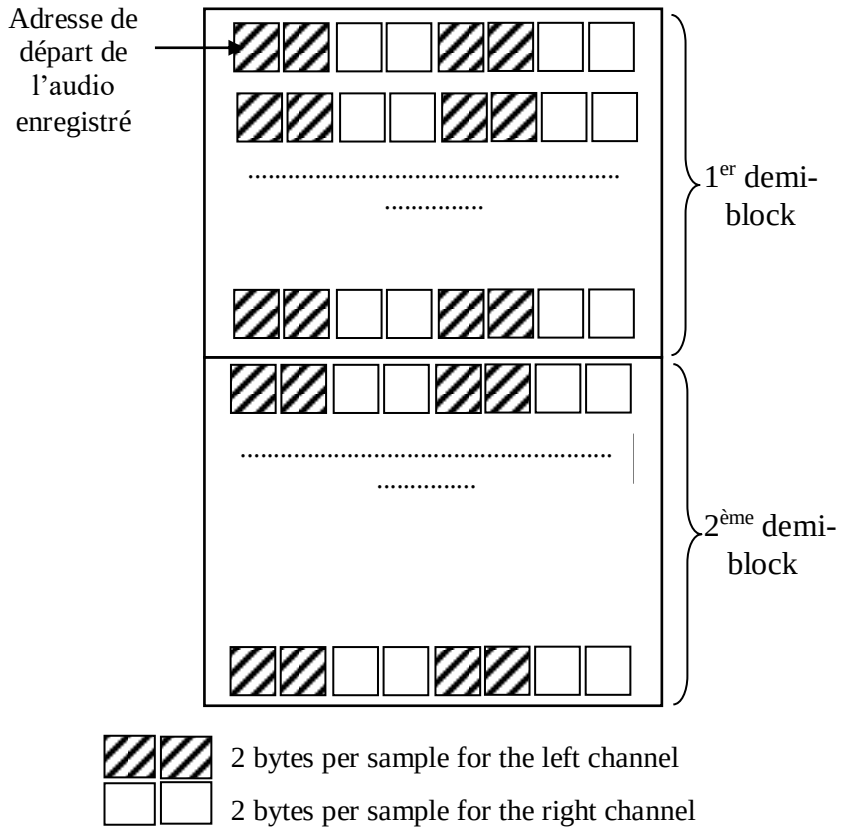


Figure 4.8: Mapping Mémoire pour un block complet (une seule trame)

Nom de variable	Désignation	Type	Nombre d'octets	Adresse de départ
T	Facteur d'échelle	Float	4	0x20000114
P	Au moins P trames vocales / non vocales acceptées	unsigned short int	2	0x20000076
Δ_E	Ecart d'énergie	Float	4	0x20000118
A	Facteur d'ajustement	Float	4	0x2000011c
Cf	Compteur des trames	unsigned long int	4	0x20000120
frm_prsnt	Buffer contenant des échantillons de la trame actuelle	Float	4*80	0x20000124
Rx	Coefficient d'autocorrélation	Float	4	0x20000264
Ef	Energie en pleine bande	Float	4	0x20000268
Z	Estimateur du niveau moyen de l'énergie du bruit de fond	Float	4*10	0x2000007c
VAD_buff	Registre des décisions partielles	unsigned short int	2*(P+1)	0x200000cc
Ef_buff	FIFO Energy-Buffer des dernières trames (P + 1)	Float	4*4	0x200000dc
E_{mm}	FIFO Energy-Buffer des dernières trames *fu*	Float	4*15	0x200000a4
E_{min}	Energie minimum dans le buffer E_{mm}	Float	4	0x20000104
E_{max}	Energie maximum dans le buffer E_{mm}	Float	4	0x20000100
Th	Seuil adaptatif	Float	4	0x20000270
Vf	Décision VAD finale	unsigned short int	2	0x20000074

Tableau 4.3: Organisation et type de données en mémoire

4.5 Résultats et discussion

4.5.1 Tests et analyse en temps réel

Afin d'évaluer le comportement en ligne de la méthode proposée à l'aide de la carte MCU, nous avons enregistré un signal audio expérimental noyé dans un environnement bruité du monde réel (babillage des étudiants). Ici, le bruit peut être considéré comme stationnaire en raison de la courte durée du signal enregistré (6 secondes). Les courbes présentées sur la Figure 4.8 ont été affichées en temps réel lors de l'enregistrement du signal via l'outil STM-STUDIO [44]. La première courbe montre l'énergie pleine bande par rapport au seuil adaptatif. On observe que le VAD est mis à 1 à chaque fois que l'énergie pleine bande dépasse le seuil adaptatif, qui correspond à une trame voisée. Notons que les sorties VAD sont illustrées en rouge sur la troisième courbe, tandis que la deuxième courbe montre le signal enregistré.

Comme mentionné précédemment, nous ne conservons que les trames déclarées comme parole et fixons les trames non vocales à 0, afin d'obtenir une évaluation subjective et de garantir que le signal n'a pas été dégradé après le traitement VAD.

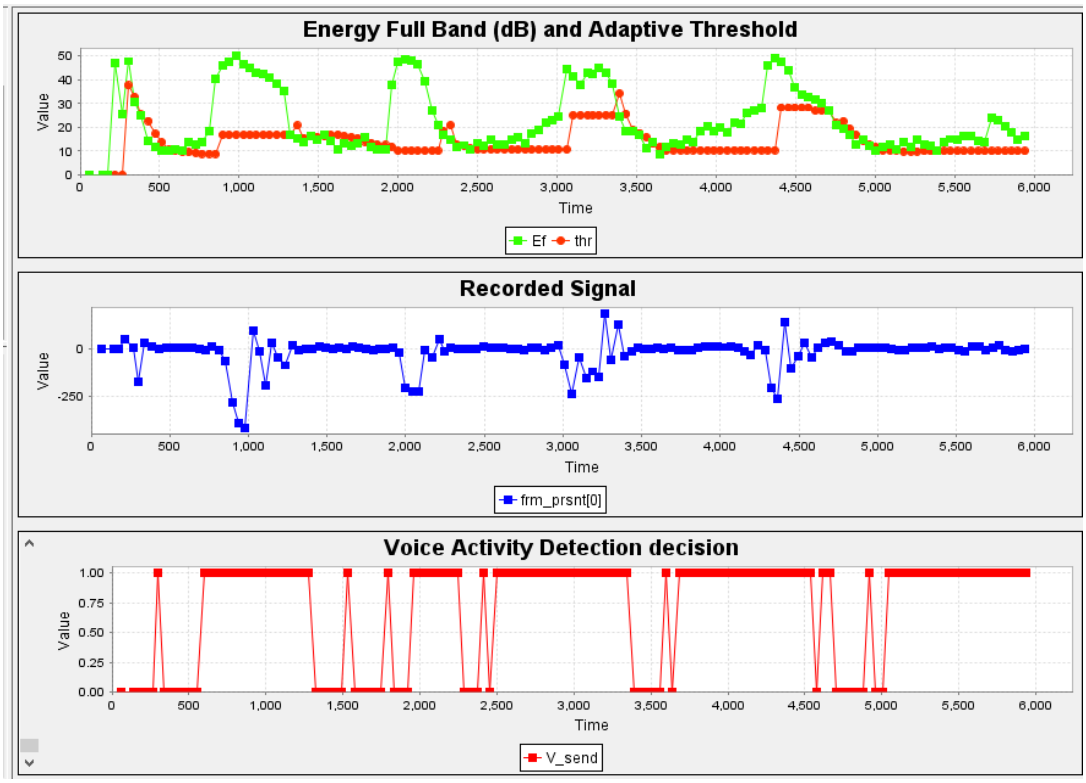


Figure 4.9: Exemple d'exécution en temps réel dans un environnement stationnaire

Pour analyser le comportement de notre le VAD dans un environnement non-stationnaire, nous avons généré un bruit blanc non stationnaire de 14 seconds, en utilisant l'outil Audacity. Pour cela, nous avons crée une situation telle que l'amplitude du bruit de fond varie de 0.1 à 04, puis décroit jusqu'à atteindre la valeur 0.1 avec un pas régulier de 0.1 ; Chacun de ces intervalles, à puissance constante, est maintenu pendant 02 secondes (Figure 4.10). Cette modification périodique de la puissance, autrement dit de la variance du processus, crée un signal aléatoire non stationnaire et simule parfaitement les conditions d'un environnement non homogène. Le début d'enregistrement du signal vocal est synchronisé avec le lancement du bruit de fond. Les résultats obtenus sont montrés dans la Figure 4.11.

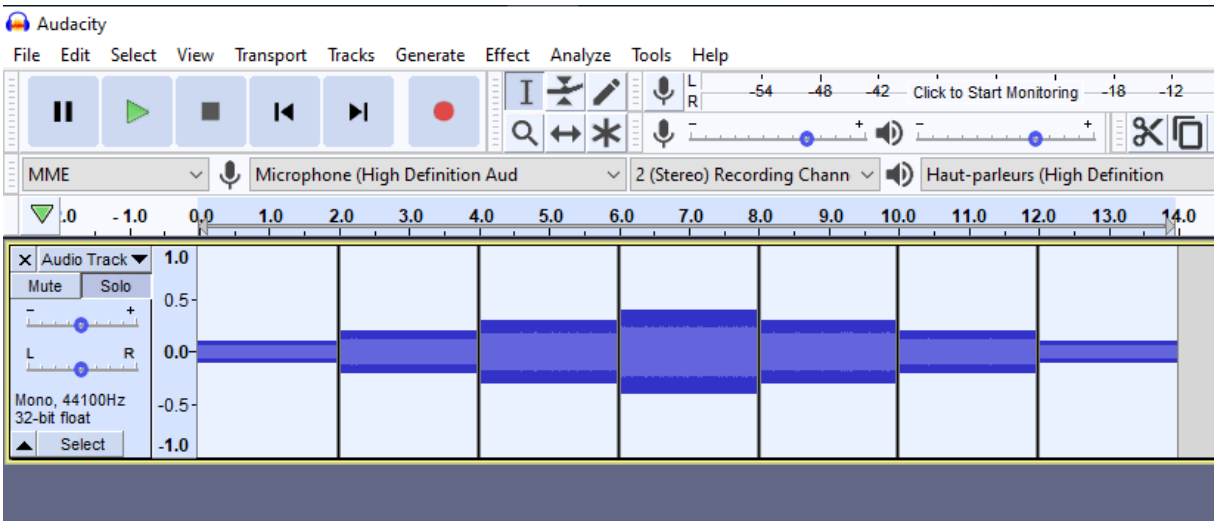


Figure 4.10: Génération du bruit de fond non-stationnaire

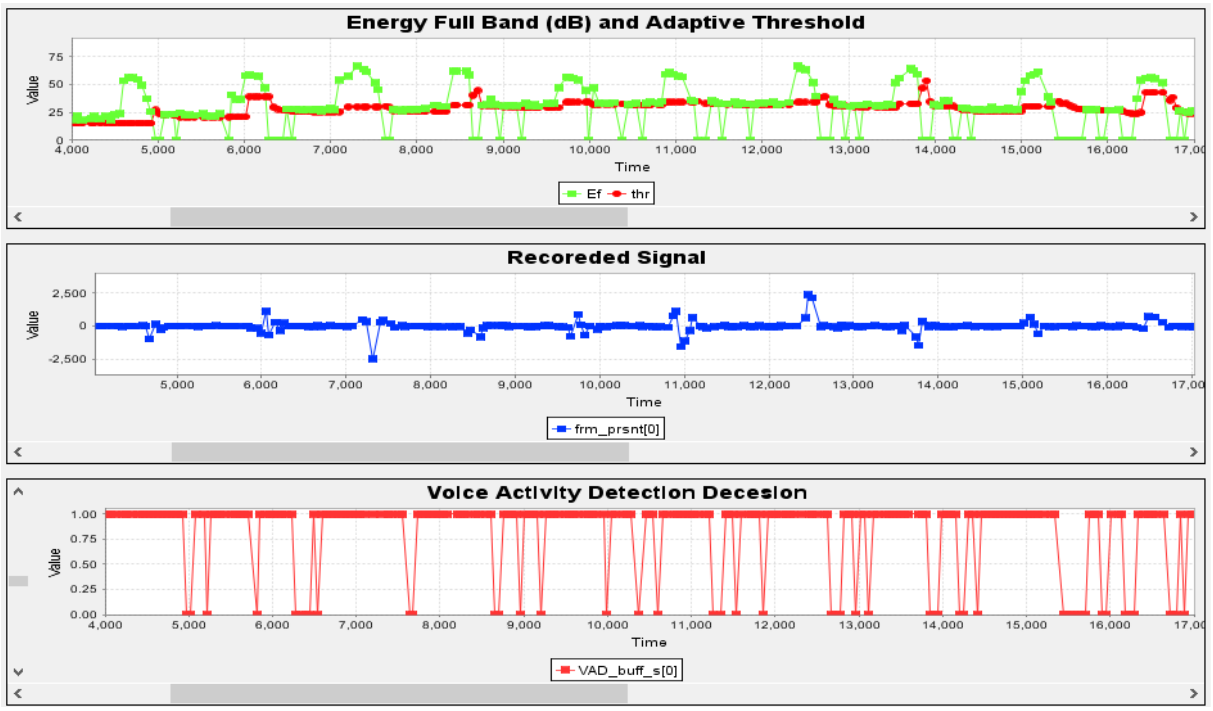


Figure 4.11 Exemple d'exécution en temps réel dans un environnement non-stationnaire

4.5.2 Exécution en temps réel

Pour valider la fonctionnalité en temps réel du système VAD proposé, nous devons analyser le comportement en ligne de l'implémentation correspondante. Pour cela, nous déterminons le temps de traitement moyen de l'algorithme VAD, qui s'exécute après chaque trame de 10 ms. La Figure 6 illustre la synchronisation d'exécution des tâches principales pour le codec et le VAD proposé, en utilisant un échantillonnage à 8 kHz (80 échantillons par trame). Le logiciel de la carte MCU fournit un registre, appelé CYCLECOUNTER, qui donne le nombre de cycles d'horloge entre deux points d'arrêt prédéfinis. Pour calculer la latence de la tâche VAD, nous avons placé deux points d'arrêt au début et à la fin du programme correspondant. Ensuite, la différence moyenne entre les deux valeurs CYCLECOUNTER successives est calculée en plusieurs essais. La moyenne du temps de traitement global du VAD s'est avérée être de 200 cycles d'horloge. En considérant une horloge CPU de 49,142 MHz [38], le temps de traitement du bloc VAD a été déterminé à environ 4 µs, ce qui est largement suffisant par rapport aux temps d'échantillonnage fréquemment utilisés dans le traitement audio (125 µs à 8 kHz ou 62,5 µs à 16 kHz).

Comme il est montré dans la Figure 4.10, après l'acquisition du dernier échantillon de chaque trame (S80, S160, S240…), le bloc complet est rempli puis passé au VAD proposé, ensuite l'algorithme s'exécute pour générer la décision finale de la trame dans un temps moyen de 4 µs.

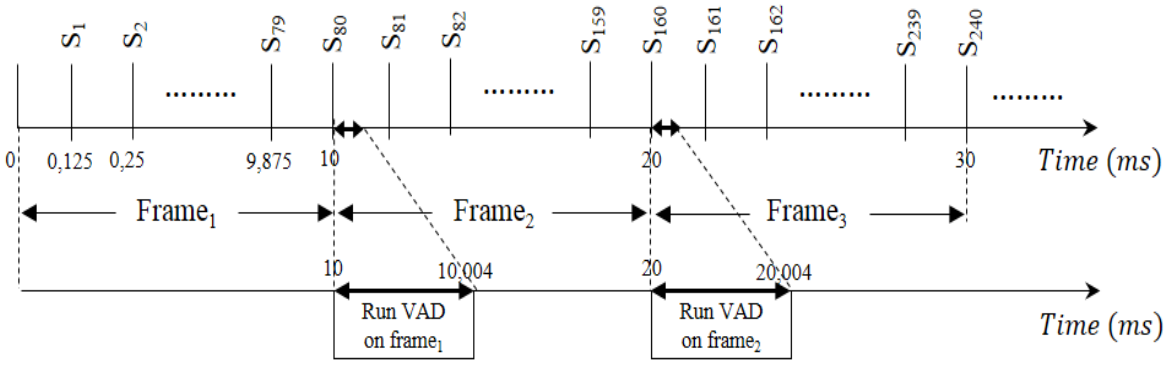


Figure 4.12: Chronogramme de la méthode proposée

4.6 Conclusion

Dans ce chapitre, nous avons présenté l'état de l'art des implémentations des méthodes VAD sur divers types de matériel (DSP, FPGA et prothèses auditives), l'architecture globale de la carte cible STM32F7 a été présentée. Pour les besoins de l'implémentation de notre technique VAD, nous avons définit les parties hardware correspondantes, ainsi que les softwares utilisés pour coder et télécharger l'application dans la carte. Ensuite, divers tests ont été effectués dans un environnement stationnaire et non stationnaire. Finalement, le temps moyen d'exécution pour générer la décision finale pour chaque trame a été estimé environ 4 µs, ce qui est largement suffisant par rapport aux fréquences d'échantillonnage utilisées dans le domaine traitement de la parole.

5-1 Conclusion :

Dans cette thèse nous avons abordé le problème de la détection de l'activité vocale (VAD) dans une communication audio avec une approche statistique dite conventionnelle. Plus particulièrement, nous nous sommes intéressés au développement d'un système VAD robuste vis-à-vis du bruit de fond stationnaire ou non-stationnaire. Ainsi, la technique proposée se devait de satisfaire les impératifs d'une application en temps réel.

Dans la 1ère contribution, nous nous somme intéressés à mettre au point une technique VAD basée sur un seuillage adaptatif, tout en maintenant une valeur nominale pour le taux de fausse acceptation (False Acceptance rate). La solution retenue est basée sur le concept de la décision binaire en présence de deux hypothèses statistiques ; en l'occurrence, l'hypothèse nulle H_0 (Silence) et l'hypothèse alternative H_1 (présence de la voix). A cette fin, nous avons considéré les distributions Gaussienne et Laplacienne pour décrire les régions du silence et celles de la voix respectivement. Dans ce processus, un seul paramètre, extrait du signal audio, a été utilisé comme variable de décision ; il s'agit de l'énergie en pleine bande de fréquence qui s'apparente au premier coefficient de l'autocorrelation. La décision finale est générée en utilisant une technique de lissage qui tient compte de l'état d'un nombre fini de trames précédent la trame sous test. Le but du lissage étant d'atténuer le phénomène de hachage dû aux discontinuités indésirables aussi bien dans les régions voisées que dans les intervalles du silence.

L'approche proposée a été testée dans un environnement stationnaire et non-stationnaire, puis comparée au VAD du standard G.729 en utilisant la base de données NOIZEUS ainsi que des signaux réels noyés dans un bruit de fond, généré par l'outil Audacity. La méthode VAD proposée a montré un comportement robuste en présence d'environnement fortement bruité. Les performances obtenues en termes de perte de détection dans les intervalles voisés (True Rejection) et en termes de taux de compression dépassent celles du G.729-B dans la plupart des situations explorées.

Dans la 2ème contribution, l'algorithme proposé a été implémenté et ciblé sur un système à base microcontrôleur pour valider son fonctionnement en temps réel, et par conséquent, afin de mettre en avant sa simplicité d'implémentation. L'approche a été testée en temps réel, analysée à l'aide des outils de monitoring en temps réel. La latence globale pour générer une décision finale a été d'environ 4us, ce qui est largement suffisant en tenant compte des

fréquences d'échantillonnage utilisées dans le domaine traitement de la parole (8 kHz à 16 kHz).

5-2 Perspectives :

Outre les approches statistiques mentionnées ci-dessus, des méthodes plus récentes ont été développées en utilisant des techniques d'apprentissage automatique [45], [46], et plus spécifiquement, en utilisant des réseaux de neurones profonds DNN (Deep Neural Network). Ces dernières années, l'utilisation des DNN est devenue un axe de recherche très prolifique notamment pour les domaines tels que la reconnaissance des formes, la classification d'images [47] ou la reconnaissance de la parole ou du locuteur [48]. Les réseaux profonds ont été utilisés avec succès pour extraire des représentations de signaux utiles à partir de données brutes, dans ce cadre, plusieurs études ont montré la puissance des réseaux profonds à modéliser la structure inhérente contenue dans le signal de parole [49].

Les réseaux DNN ont été récemment utilisés dans plusieurs détecteurs VAD modernes de réponse vocale; à titre d'exemple, Zhang et Wu [50] ont proposé d'extraire un ensemble de caractéristiques acoustiques prédéfinies, à partir d'un signal de parole, puis de transmettre ces caractéristiques à un réseau de croyances profondes DBN (Deep-Belief Network) afin d'obtenir une représentation plus significative du signal.
Ils ont ensuite utilisé un classificateur linéaire pour effectuer la détection de la parole.

Des méthodes plus modernes reposent sur des réseaux de neurones récurrents RNN (Recurrent neural network) pour incorporer les entrées précédentes dans le processus de classification, utilisant ainsi les informations temporelles du signal [51], [52], [53]. Dans [54] Lim et coll ont proposé de transformer le signal de parole en utilisant une transformée de Fourier à court terme puis d'utiliser un CNN (Convolutional Neural Network) pour extraire une représentation de haut niveau pour le signal.

Bibliographie

[1] Matras, J. J. « Le son (Que sais-je) ». (1948).

[2] Mc Gill. http://www.lecerveau.mcgill.ca/flash/capsules/outil_bleu21.html, (Consulté le 24 Avril 2021).

[3] Schafer, R. W., & Markel, J. D. (1979). Speech analysis. New York: IEEE Press.

[4] http://www.medecine-et-sante.com/anatomie/anatoreille.html, (Consulté le 24 Avril 2021).

[5] Tetschner, W. (1993). "Voice processing". Boston: Artech House

[6] Parsons, T. (1986). "Voice and speech processing". New York: Mc Graw-Hill

[7] Oppenheim, A. V., & Schafer, R. W. (1989). Discrete-time signal processing. New Jersey: Prentice Hall.

[8] Freeman, D.K.; Cosier, G.; Southcott, C.B.; Boyd, I. (1989). "The Voice Activity Detector for the PAN-European Digital Cellular Mobile Telephone Service", International Conference on Acoustics, Speech and Signal Processing, Vol. 1, pp. 369-372

[9] Sangwan, A.; Chiranth, M.C.; Jamadagni, H.S.; Sah, R.; Prasad, R.V.; Gaurav, V. (2002). "VAD Techniques for Real-Time Speech Transmission on the Internet", IEEE International Conference on High-Speed Networks and Multimedia Communications, pp. 46-50.

[10] Itoh, K.; Mizushima, M. (1997). "Environmental noise reduction based on speech/non-speech identification for hearing aids", International Conference on Acoustics, Speech, and Signal Processing, Vol. 1, pp. 419-422.

[11] ITU, "A silence compression scheme for G.729 optimized for terminals conforming to ITU-T V.70", ITU-T Rec. G. 729, Annex B, 1996.

[12] ETSI, Voice activity detector (VAD) for adaptive multirate (AMR) speech traffic channels, ETSI EN 301 708 v.7.1.1, Dec. 1999.

[13] ETSI. (1998). Digital cellular telecommunications system (Phase 2+); Voice Activity Detector (VAD) for adaptative Mufti-Rate (AMR) speech traffic channels; General description, GSM, 06.94 version 7.1.1 release 1998.

[14] Vahatalo, A., & Johansson, I. (1999). "Voice activity detection for GSM adaptive multi-rate code". Speech Coding Proceedings, 1999 IEEE Workshop on, pp. 55-57.

[15] ETSI, “Speech processing, transmission and quality aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms,” ETSI, Sophia Antipolis, France, ETSIES 202 050 Rec., 2002.

[16] Skype. Silk speech codec, 2009. Accessed 14 April 2011; http://developer.skype.com/silk

[17] Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator,” IEEE Trans. Acoust., Speech, Signal Process., vol. 32, no. 6, pp. 1109–1121, Dec. 1984.

[18] J. Sohn and W. Sung, “A voice activity detector employing soft decision based noise spectrum adaptation,” Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., pp. 365–368, 1998

[19] J. Sohn, N. S. Kim, and W. Sung, “A statistical model-based voice activity detection,” IEEE Signal Process. Lett., vol. 6, no. 1, pp. 1–3, Jan. 1999.

[20] R. Martin, “Speech enhancement using MMSE short time spectral estimation with gamma distributed speech priors,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., vol. 1, Orlando, FL, May 2002, pp. I253–I256.

[21] S. Gazor and W. Zhang, “Speech probability distribution,” IEEE Signal Process. Lett., vol. 10, pp. 204–207, Jul. 2003.

[22] I.Cohen, “Noise estimation by minima controlled recursive averaging for robust speech enhancement,” IEEE Signal Process. Lett., vol. 9, pp. 12–15, Jan. 2002.

[23] R. Martin, “Noise power spctral density estimation based on optimal smoothing and minimum statistics,” IEEE Trans. Speech Audio Process., vol. 9, pp. 504–512, Jul. 2001.

[24] J. Chang et al, “Voice activity detection based on multiple statistical models”, IEEE Transactions on Signal Processing, vol. 54, no 6, pp. 1965 - 1976, June 2006.

[25] R. C. Reininger and J. D. Gibson, “Distributions of the two dimensional DCT coefficients for images,” IEEE Trans. Commun., vol. COM-31, no. 6, pp. 835–839, Jun. 1983.

[26] E. Etellisi, P. Papantoni-Kazakos, “Sequential Tests for the Detection of Voice Activity and the Recognition of Cyber Exploits,” SciRes, Communications and Network, vol. 3,no. 4,pp. 185-199, November 2011.

[27] R. Muralishankar, R. Venkatesha Prasad, S. Vijay and H. N. Shankar, “Order Statistics for Voice Activity Detection in VoIP,” Proc. IEEE International conference on communications, pp. 1-6, May 2010.

[28] Papoulis A, “Probability, random variables and stochastic processes”, MC Graw-Hill. 3rd edition. 1991.

[29] Y. Hu., P. C. Loizou. ”Subjective evaluation and comparison of speech enhancement algorithms”, IEEE International Conference on Acoustics Speech and Signal Processing Proceedings., July 2006, pp. 588-601.

[30] H. G. Hirsch., D. Pierce.:'The AURORA experimental framework for the performance evaluation of speech recognition systems under noise conditions', Proc. ISCA ITRW ASR Challenges for the Next Millennium, September 2000.

[31] IEEE Subcommittee, “IEEE recommended practice for speech quality measurements,” IEEE Trans. Audio and Electroacoustics, pp. 225–246, 1969.

[32] S. Mousazadeh., I. Cohen. ”Voice activity detection in presence of transient noise using spectral clustering”, IEEE Transactions on Audio, Speech, and Language Processing,2013, 21, (6), pp. 1261-1271

[33] Audacity® software is copyright © 1999-2021 Audacity Team. Web site: https://audacityteam.org/. It is free software distributed under the terms of the GNU General Public License. The name Audacity® is a registered trademark.

[34] M. Oukherfellah and M. Bahoura, “FPGA implementation of voice activity detector for efficient speech enhancement,” in IEEE 12th International New Circuits and Systems Conference, June 2014, pp. 301–304.

[35] Liang, J., Ahmad, M. O., & Swamy, M. N. S. (2005). “Implementation of a voice activity detection and comfort noise generation Algorithm”. In 48th Midwest Symposium on Circuits and Systems, Vol. 1, pp. 440–443

[36] L. Narimene, et al. 2014. “Voice activity detection system for smart earphones”. IEEE Transactions on Consumer Electronics, vol. 60, nº 4. pp. 737-744.

[37] K. Mazur, J. Voix, “Implementing 24-hour in-ear dosimetry with recovery” in Proc. International Conference on Acoustics, New York, USA, 2013.

[38] STM32F746-Disco, https://www.st.com/en/evaluation-tools/32f746gdiscovery.html

[39] Codec WM8994, https://www.cirrus.com/products/wm8994

[40] Liste des Environnements de Développement Intégrés, https://www.st.com/en/development-tools/stm32-ides.html#2

[41] Liste des softwares d’analyse en temps réel, https://www.st.com/en/development-tools/stm32-performance-and-debuggers.html#2

[42] STM32CubeMX, https://www.st.com/en/development-tools/stm32cubeide.html#get-software

[43] IAR Workbensh, https://www.st.com/en/development-tools/iar-embedded-workbench-for-arm.html

[44] STM-STUDIO, https://www.st.com/en/development-tools/stm-studio-stm32.html

[45] Jong Won Shin, Joon-Hyuk Chang, and Nam Soo Kim, “Voice activity detection based on statistical models and machine learning approaches,” Computer Speech & Language, vol. 24, no. 3, pp. 515–530, 2010.

[46] Ji Wu and Xiao-Lei Zhang, “Maximum margin clustering based statistical VAD with multiple observation compound feature,” IEEE Signal Processing Letters, vol. 18, no. 5, pp. 283–286, 2011.

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A Large-Scale Hierarchical Image Database,” in Proc. CVPR09, 2009

[48] John S Garofolo, “Timit acoustic phonetic continuous speech corpus,” Linguistic Data Consortium, 1993.

[49] Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, et al., “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” IEEE Signal processing magazine, vol. 29, no. 6, pp. 82–97, 2012.

[50] Xiao-Lei Zhang and Ji Wu, “Deep belief networks based voice activity detection,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 4, pp. 697–710, 2013.

[51] Simon Leglaive, Romain Hennequin, and Roland Badeau, “Singing voice detection with deep recurrent neural networks,” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 121–125.

[52] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, “Speech recognition with deep recurrent neural networks,” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 6645–6649.

[53] Wei-Tyng Hong and Chien-Cheng Lee, “Voice activity detection based on noise-immunity recurrent neural networks,” International Journal of Advancements in Computing Technology (IJACT), vol. 5, no. 5, pp. 338–345, 2013

[54] Wootaek Lim, Daeyoung Jang, and Taejin Lee, “Speech emotion recognition using convolutional and recurrent neural networks,” in Proc. Signal and Information Processing Association Annual Summit and Conference (APSIPA). IEEE, 2016, pp. 1–4.