MÉMOIRE DE STAGE ATIAM

La notion d'effort dans la synthèse vocale performative

INSTITUT JEAN LE ROND ∂'ALEMBERT

ÉQUIPE LUTHERIES, ACOUSTIQUE, MUSIQUE

Auteur :
Grégoire LOCQUEVILLE

Encadrement :
Christophe d'ALESSANDRO
Boris DOVAL

Février 2018 — Juillet 2018
20 août 2018

Abstract

Dans une perspective artistique, mais aussi thérapeutique, il peut être utile pour un être humain d'externaliser la voix, c'est-à-dire d'être en mesure de faire entendre une voix générée en temps réel sans faire usage de son conduit vocal. C'est la raison d'être des *instruments chanteurs*, des synthétiseurs vocaux en temps réel. Une problématique importante dans l'utilisation de ces instruments est celle des *variations expressives* de la voix. Il s'agit alors de trouver à la fois des manières de faire varier les émotions transmises par une voix, et des manières de contrôler ces variations.

Dans cette optique, l'utilisation d'un logiciel permettant de coder et modifier la voix, le vocodeur World, s'avère appropriée. Au cours de ce stage, son fonctionnement a été étudié et compris en détail, et le vocodeur a été modifié pour permettre la prise en compte de données en temps réel issues de capteurs externes, et la modification de la voix traitée.

Un nouvel instrument chanteur, Voks, a alors été conçu et développé. Il permet, en s'appuyant sur un enregistrement de voix préexistant, de le rejouer avec n'importe quel rythme, sur n'importe quelle mélodie, en introduisant des variations expressives, le tout en temps réel. Un soin particulier a été apporté aux variations d'un paramètre à la fois naturel et complexe, l'effort vocal.

Les résultats audio produits par Voks ont finalement été informellement évalués pour divers exemples de voix, et les différents éléments en jeu dans l'effort vocal, comparés grâce à cet instrument.

Mots-clés : synthèse vocale performative, expression, effort vocal, vocodeur

Abstract

Externalizing one's voice, that is, being able, without making use of one's vocal tract, to make heard a voice that has been generated in real-time, can be seen as useful in artistic contexts as well as therapeutic ones. That is the reason why a number of *singing instruments*, real-time vocal synthesizers, have been developed. An important question, when making one of those instruments, is how to implement and control *expression variations* of the voice; in other words, one has to devise ways both to vary the emotions conveyed by a voice, and to control those variations.

To that end, the World vocoder has proven useful. World is a piece of software that allows for coding and modification of voice. During this internship, its inner workings have been extensively studied and understood, and it has been modified so as to be able to take external sensor data into account, and to modify the considered voice.

A new singing instruments, Voks, has then been designed and developed. Voks allows, using a preexisting recording of a voice, to play it back in real time, with any rhythm, any melody, and with expression variations. A specific parameter, both natural and complex, has been considered: vocal force.

Voks' audio results have then been informally evaluated on several voice recordings of diverse origins, and the elements that come into play in the perception of vocal force have been compared using that instrument.

Keywords: performative vocal synthesis, expression, vocal force, vocoder

Remerciements

Je souhaite avant tout remercier mes encadrants, Boris Doval et Christophe d'Alessandro, de m'avoir permis de travailler sur un sujet complet et passionnant, et pour leur aide durant ce stage.

J'aimerais également remercier les personnes que j'ai côtoyées et avec qui j'ai eu l'occasion de travailler : les membres du LAM, y compris les autres stagiaires, avec qui j'ai partagé de bons moments, et notamment Xiao Xiao, avec qui j'ai pu explorer de nouvelles manières de faire de la musique.

Enfin, je remercie l'équipe pédagogique d'ATIAM, qui m'a permis d'apprendre beaucoup durant cette année, et qui m'a également accompagné durant ce stage — notamment ceux de ses membres qui font partie du LAM.

Table des matières

Introduction **4**
Présentation du LAM 4
Sujet du stage 4

I Problématiques scientifiques **6**
I.1 Un modèle phénoménologique de la voix : le modèle source-filtre . . 6
I.1.1 La source 7
I.1.2 Le filtre 7
I.1.3 Le vocoding, une application du modèle source-filtre 8
I.2 Variations expressives de la voix 9
I.2.1 Variations segmentales (aspects signal) 10
I.2.2 Variations suprasegmentales (aspects prosodiques) 11
I.3 Vokinesis 13
I.3.1 Principe général 13
I.3.2 Modalités de contrôle du rythme 14

II Le vocodeur World **16**
II.1 Fonctionnement de World 16
II.1.1 Analyse 16
II.1.2 Synthèse 22
II.2 Aspects pratiques 25
II.2.1 Limites 25
II.2.2 Exploitation du vocodeur 25

III L'instrument Voks **28**
III.1 Technologies utilisées 28
III.2 Utilisation de l'instrument 29
III.2.1 Interfaces de contrôle 30
III.3 Structure 32
III.3.1 Calcul du temps d'analyse 33
III.3.2 Extraction des coefficients World 34
III.3.3 Calcul des paramètres expressifs 35
III.3.4 Transformation expressive des coefficients World 36

III.3.5 Synthèse sonore 37
III.4 Modifications expressives de la voix synthétisée 38
III.4.1 Variation de la quantité d'apériodicité 38
III.4.2 Variation de l'équilibre première harmonique / harmoniques supérieures 38
III.4.3 Pente spectrale 39
III.4.4 Gain 40
III.4.5 Limites 40

IV Applications **42**
IV.1 Élaboration d'exemples sonores 42
IV.1.1 Enregistrements 42
IV.1.2 Étiquetage 43
IV.1.3 Jeu 44
IV.1.4 Génération d'exemples sonores avec variation des paramètres 44
IV.2 Résultats 44
IV.A Liste des modifications de paramètres effectuées 45

Conclusion **48**
Travail réalisé 48
Travail restant, perspectives 48

Introduction

Présentation du LAM

Né en 2007 du rassemblement de cinq laboratoires, l'Institut Jean le Rond ∂'Alembert accueille des recherches en acoustique, mécanique et énergétique. Parmi les cinq équipes qui le composent, l'équipe *Lutheries, Acoustique, Musique* (héritière du Laboratoire d'Acoustique Musicale fondé en 1963) s'attache à l'étude de l'objet musical selon une approche qui mêle des disciplines scientifiques et techniques telles que l'acoustique et le traitement du signal, et des sciences humaines comme la psycholinguistique. C'est ainsi que les recherches qui y sont menées se font fréquemment en collaboration avec des musiciens et des luthiers. Ce positionnement original en fait une équipe pluridisciplinaire tant en ce qui concerne les problématiques abordées que du point de vue institutionnel : l'équipe LAM est ainsi sous la tutelle conjointe du CNRS, de Sorbonne Université et du Ministère de la Culture.

Sujet du stage

Depuis plusieurs années, le LIMSI (Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur) et le LAM développent des outils de synthèse de voix contrôlée en temps réel par des humains, ou *synthèse vocale performative*. Ces instruments de musique, auxquels a été donné le nom d'*instruments chanteurs*, permettent *via* le geste humain d'introduire un aspect organique qui est pour beaucoup dans le réalisme et l'expressivité de la voix synthétisée.

Pour que la synthèse soit vraiment réaliste, un paramètre, l'*effort vocal*, est incontournable. S'il s'agit d'une notion très intuitive, elle s'avère difficile à formaliser. Ce stage vise à mieux comprendre les mécanismes qui entrent en jeu dans la modification de l'effort vocal, et à les mettre en œuvre dans un nouvel instrument chanteur. Pour ce faire, un outil, le vocodeur World, qui permet la modification et la synthèse en temps réel d'un signal de voix, nous a été précieux.

Le premier chapitre de ce document rappelle les problématiques scientifiques qui sont entrées en jeu lors de ce stage, et fournit les informations nécessaires à la compréhension de la suite du rapport.

La deuxième partie est le fruit d'un travail de compréhension du vocodeur World, mal documenté. Cette compréhension en profondeur du vocodeur a permis de le modifier, pour pouvoir l'interfacer avec le langage audio temps réel Max et le faire prendre en compte des données issues d'interfaces de contrôle.

Grâce ce travail de modification et d'interfaçage, la création d'un nouvel instrument chanteur ayant World pour moteur audio est devenue envisageable. S'est alors ensuivie la conception d'un tel instrument, Voks, décrit dans le troisième chapitre : définition des fonctionnalités et de la manière de les implémenter, test de différentes interfaces de contrôle. L'instrument Voks a finalement été implémenté en Max.

Enfin, en dernière partie sont décrites l'évaluation musicale (informelle) de cet instrument et son utilisation pour évaluer l'importance relative de différents paramètres dans la perception de l'effort vocal.

Chapitre I

Problématiques scientifiques

Ce stage fait appel à des connaissances issues de disciplines diverses, du traitement du signal à la phonogie. Dans ce chapitre, on rappelle celles de ces connaissances qui sont nécessaires pour la compréhension du travail réalisé au cours de ce stage.

I.1 Un modèle phénoménologique de la voix : le modèle source-filtre

Un modèle en synthèse de voix est très utilisé en raison de son efficacité et de sa simplicité. Décrit dans [11], dont nous reprenons ici les notations, le modèle source-filtre est un modèle *phénoménologique* de la production de parole. Autrement dit, il ne s'appuie pas sur une modélisation physique, même simplifiée, de l'appareil vocal, mais sur l'observation directe du son produit par ledit appareil ; et si, dans de nombreuses descriptions du modèle, on s'autorise à dire, par exemple, que « la source correspond aux vibrations des plis vocaux », il s'agit plus de l'expression d'une représentation mentale commode que d'une réalité physique précise.

Comme son nom l'indique, le modèle source-filtre suppose que la voix est le résultat du *filtrage* d'un signal riche en harmoniques — la *source* —, qui correspond à la vibration des plis vocaux (les « cordes vocales »), par le conduit vocal, constitué de plusieurs cavités, notamment le larynx et la bouche. L'onde de pression produite $p(t)$ s'écrit alors, dans le domaine fréquentiel :

$$P(f) = S(f)T(f)$$

où $S(f)$ correspond à la source, et $T(f)$ est la réponse en fréquence du filtre. Les sections suivantes décrivent les caractéristiques de ces deux éléments.

I.1.1 La source

La source comporte toujours au moins une partie périodique, à laquelle vient en général s'ajouter du bruit. Dans le domaine fréquentiel, un signal de période T_0 correspondant à des pics aux multiples de $\frac{1}{T_0}$, la source s'écrit alors :

$$s(t) = \left(\sum_{k \in \mathbb{Z}} c_k \delta \left(t - \frac{k}{T_0} \right) \right) + n(t)$$

où :

- les c_k sont des coefficients (complexes) dépendant des caractéristiques de la source,
- $n(t)$ est un bruit, blanc pour des modèles simples, ou coloré,
- δ est la fonction de Dirac.

La partie périodique de la source correspond à l'ouverture et la fermeture régulière des plis vocaux, tandis que la partie apériodique correspond aux phénomènes turbulents qui se produisent à ce niveau. La partie apériodique permet également de prendre en compte du bruit produit dans le conduit vocal, notamment lors de la prononciation de consonnes. Une description plus précise de la partie périodique de la source est faite dans [8], où est dégagé ce que différents modèles de la source ont de commun.

I.1.2 Le filtre

Les caractéristiques du filtre sont liées à la configuration du conduit vocal, c'est-à-dire de cavités telles que le larynx, la bouche, et parfois les fosses nasales. Cette configuration, bien entendu, dépend de l'anatomie propre du locuteur, mais un locuteur seul peut également la modifier en jouant sur les différents éléments qui composent le conduit vocal : hauteur du larynx, position de la langue, des lèvres, ouverture/fermeture des fosses nasales... On peut ainsi faire varier les pics qui apparaissent sur la réponse en fréquence du filtre, appelés *formants*.

Les formants

La réponse en fréquence du filtre formé par le conduit vocal possède des pics, dont l'amplitude, et surtout la fréquence, sont caractéristiques des phonèmes prononcés. Notons toutefois que ces valeurs dépendent aussi du contexte de prononciation : phonèmes précédent et suivant, morphologie du locuteur... Malgré tout, les

fréquences de ces formants (notées F_1, F_2, *etc.* — f_0 étant utilisé pour dénoter la fréquence fondamentale), notamment celles des premiers, donnent une certaine quantité d'informations sur le phonème prononcé [15], et on peut trouver dans la littérature [7] des tableaux comme celui en figure I.1 répertoriant leurs valeurs pour différents phonèmes, pour différents types de locuteurs.

	Male			Female		
	F_1	F_2	F_3	F_1	F_2	F_3
i:	280	2249	2765	303	2654	3203
ɪ	367	1757	2556	384	2174	2962
e	494	1650	2547	719	2063	2997
æ	690	1550	2463	1018	1799	2869
ʌ	644	1259	2551	914	1459	2831
ɑ:	646	1155	2490	910	1316	2841
ɒ	558	1047	2481	751	1215	2790
ɔ:	415	828	2619	389	888	2796
ʊ	379	1173	2445	410	1340	2697
u:	316	1191	2408	328	1437	2674
ɜ:	478	1436	2488	606	1695	2839

FIGURE I.1 – Valeur moyenne de la fréquence en Hz des premier (F_1), deuxième (F_2) et troisième (F_3) formants pour les voyelles de la langue anglaise. Tableau tiré de [7]

C'est sur ces formants que l'on joue, en synthèse, pour produire des voyelles précises, comme c'est le cas par exemple dans le Cantor Digitalis [13]. Mais on peut également changer ces formants, entre autres manipulations, en vue de jouer sur des aspects expressifs de la voix, comme décrit en section I.2.1.

I.1.3 Le vocoding, une application du modèle source-filtre

C'est en s'appuyant sur le modèle source-filtre que l'ingénieur Homer Dudley a développé l'idée du *vocodeur* [9]. Conçu à l'origine pour réduire la bande passante nécessaire dans les télécommunications, le vocodeur a depuis trouvé des applications dans d'autres domaines.

Originellement, l'idée du vocodeur consiste à déterminer une approximation du filtre du signal initial $e(t)$ (typiquement, la voix d'un interlocuteur au téléphone) en le faisant passer par un certain nombre n de filtres, et en transmettant uniquement les valeurs du niveau sonore du signal à la sortie de chacun de ces filtres : $\forall i = 1, 2, \ldots, n$,

$$v_i(t) = \sqrt{\int_{t-\frac{\Delta t}{2}}^{t+\frac{\Delta t}{2}} |e * h_i(t)|^2 \, \mathrm{d}t} \tag{I.1}$$

où

- $e(t)$ est le signal en entrée ;
- $h_i(t)$ ($i = 1, \ldots, n$) sont les réponses de chacun des filtres choisis ;
- $v_i(t)$ ($i = 1, \ldots, n$) sont les signaux en sortie de l'analyse — ceux qui seront transmis ;
- Δt est la taille de la fenêtre d'analyse choisie. Elle doit être grande devant la période fondamentale du signal initial, mais petite devant les changements de phonèmes et les variations expressives du locuteur.

Pour reconstituer un signal à l'autre bout de la ligne, on fait passer un signal $b(t)$ au spectre riche (un « buzz », ou même un simple bruit blanc, ou encore la somme des deux) dans chacun des filtres, pondérés par la valeur v_i transmise :

$$s(t) = \sum_{i=1}^{n} v_i(t) \, (b * h_i(t)) \tag{I.2}$$

avec

- $b(t)$ est un signal riche en harmoniques ;
- $s(t)$ est le signal en sortie.

Le signal obtenu est généralement intelligible lorsque le nombre de filtres est suffisant, ce qui a fait de cette méthode un outil utile dans les télécommunications. Le vocodeur trouve aujourd'hui des applications artistiques : en utilisant pour $b(t)$ lors de la reconstitution I.2 un signal musical, on entend un son musical qui a certaines caractéristiques de la voix, souvent comparé à une « voix de robot ». Il est également utilisé dans le domaine médical : dans de nombreux implants cochléaires, le son reçu de l'extérieur est traité comme en I.1, et les amplitudes $v_i(t)$ correspondantes sont utilisées pour stimuler différentes zones de la cochlée en envoyant des impulsions électriques.

I.2 Variations expressives de la voix

Pour offrir à l'utilisateur d'un instrument chanteur des moyens d'expression, de nouvelles manipulations du signal vocal sont nécessaires. On s'intéresse ici spécifiquement aux manipulations susceptibles de changer la perception de l'effort vocal.

Dans [29] est proposée une définition de l'effort vocal comme « la quantité que font varier des locuteurs ordinaires lorsqu'ils adaptent leur discours à une variation de distance de communication ». On pourrait adapter cette définition selon les protocoles expérimentaux : ainsi, dans [1], c'est la quantité modifiée par un locuteur lorsqu'il cherche à atteindre un certain niveau sonore. Dans tous les cas, l'effort vocal est la quantité sous-jacente que change un locuteur qui cherche à répondre à

certaines contraintes ; les variations de cette quantité se manifestent par des variations d'autres paramètres mesurables.

Certains de ces paramètres, comme ceux étudiés en I.2, ont un sens à une échelle temporelle plus petite que celle de la syllabe ; c'est notamment le cas des paramètres liés au spectre du son. Pour cette raison, ils sont qualifiés de *segmentaux*, par opposition aux paramètres *suprasegmentaux*, qui concernent plusieurs syllabes et leurs relations.

I.2.1 Variations segmentales (aspects signal)

		Relative Sound Pressure Level of Vowels (db.)			*Fundamental Frequency of Vowels* (cps.)			*Relative Sound Pressure Level of Phrases* (db.)			*Fundamental Frequency of Phrases (pooled)* (cps.)	*Duration of Phrases* (sec.)		
		[α]	[o]	[i]	[α]	[o]	[i]	1	2	3		1	2	3
Soft:	0 db.	0.9	0.9	0	109.7	109.7	110.2	0.9	0.1	0	111.8	1.32	1.33	1.12
Plus:	10 db.	14.6	11.8	8.2	121.8	123.0	125.2	10.0	8.2	7.9	129.2	1.26	1.22	1.00
Plus:	20 db.	23.7	21.1	17.5	156.2	160.6	166.2	19.2	17.3	16.7	167.4	1.29	1.21	1.02
Plus:	30 db.	32.9	33.2	27.4	222.7	233.5	250.9	29.0	27.6	25.5	237.1	1.31	1.20	0.99

FIGURE I.2 – Variation de différents paramètres lors d'une variation d'effort vocal lorsqu'il est demandé à un locuteur de prononcer une phrase normalement, puis de manière à atteindre un niveau sonore de 10, 20 et 30 dB plus élevé. Tableau tiré de [1]

Niveau sonore

Le paramètre ayant une influence sur l'effort vocal le plus évident est le niveau sonore ; l'effort vocal pourrait même être défini, en modifiant légèrement la définition de [29], comme le paramètre qui varie lorsqu'un locuteur cherche à changer le niveau sonore de sa voix. Comme il est souligné en [29], toutefois, le niveau sonore est loin d'être le seul paramètre qui change lorsque varie cet effort : un locuteur cherchant uniquement à modifier le niveau sonore de sa voix, mais qui ne cherche pas spécifiquement à laisser constants les autres paramètres, agira nécessairement sur ces derniers. Le niveau sonore n'est donc pas suffisant pour rendre compte de l'effort vocal, *a fortiori* dans le cadre d'une voix de synthèse, dont on peut régler le volume par simple pression d'une touche.

Apériodicité

En reprenant les notations de la section I.1, on peut décomposer une voix comme la somme d'une partie périodique et d'une partie apériodique $w(t) * h_a(t)$. Le poids relatif de ces deux parties est exprimé par le rapport $R_a(\omega)$. Dans [4], il est proposé

de modifier ce rapport en fonction de différents régimes de voix, correspondant à des efforts vocaux différents.

Rythme

Si les variations d'effort vocal sont susceptibles d'agir sur des aspects « signal » de la voix, des notions de plus haut niveau, tel le rythme, sont également susceptibles d'être impliquées. Ainsi, dans [1], il est noté que la vitesse d'élocution est modifiée avec l'effort vocal : les phrases prononcées le moins fort le sont aussi le moins rapidement.

Fréquence fondamentale

Toujours dans [1] (voir figure I.2), ainsi que dans dans [16], le profil de fréquence fondamentale de la voix est modifié lorsqu'augmente l'effort vocal, avec un saut de plus d'une octave entre les niveaux extrêmes d'effort vocal.

Modification des formants

Comme on pourrait s'y attendre, les amplitudes individuelles des formants augmentent avec l'intensité globale de la voix [16]. On remarque également une autre tendance : à l'instar de la fréquence fondamentale f_0, la fréquence du premier formant F_1 augmente avec l'effort vocal.

Répartition de l'énergie

Dans [32], il est noté que l'énergie d'une voix est répartie plutôt dans les basses fréquences lorsque l'effort vocal est minimal, et qu'elle se trouve de plus en plus dans les hautes fréquences à mesure qu'il augmente. Ceci suggère l'utilisation d'un filtre passe-bas pour diminuer l'effort vocal, comme il est fait dans [26]. Pour augmenter cet effort, les auteurs de [26] font appel à une distortion, qui, contrairement au filtre passe-bas, augmente l'énergie dans les hautes fréquences.

I.2.2 Variations suprasegmentales (aspects prosodiques)

Pour rendre le contrôle d'une voix de synthèse intuitif, il convient de comprendre les mécanismes en jeu dans l'articulation de la parole et comment les exploiter dans le cadre de la synthèse vocale en français. La notion de *centre perceptif* permet de comprendre comment les phonèmes s'articulent pour former un rythme, mais sa seule connaissance n'est pas suffisante pour rendre compte de la richesse d'une articulation. La théorie *frame/content* offre un cadre cognitif à la production de parole susceptible d'être exploité en synthèse vocale performative ; en combinant les idées liées à ces deux concepts, on parvient à la notion de *points de contrôle*. Les points de

contrôle permettront un contrôle expressif de l'articulation, adapté à une utilisation dans le cadre d'un instrument chanteur.

Centre perceptif

Pour définir le rythme d'une énonciation, il faut disposer de points temporels précis où l'on considère que s'effectue un changement de syllabe. Une manière naïve de faire serait de définir un changement de syllabe à la limite entre deux phonèmes telle que calculée par un logiciel tel que [2], mais il a été montré dans [25] que cette méthode ne correspond pas exactement à ce qui est perçu. Les auteurs de [25] proposent donc la notion de *p-center* (centres perceptifs), qui correspondent au point où se produit la syllabe. Si ce point se trouve effectivement vers le début de la syllabe, un certain laps de temps s'écoule généralement entre le début du son et le p-center.

La théorie frame/content

En s'appuyant sur des arguments linguistiques, anatomiques et neurobiologiques, MacNeilage fait l'hypothèse [20] que la parole repose sur deux mécanismes cognitifs distincts :

- Le geste cyclique d'ouverture et de fermeture du conduit vocal, évolution du geste de mastication, découpe la parole en syllabes et fait alterner consonnes et voyelles, constituant un cadre (en anglais : *frame*) pour les gestes plus fins requis par la parole ;
- les multiples subtils mouvements des différentes parties du conduit vocal sont le contenu (en anglais *content*) qui s'inscrit dans ce cadre, et sont rendus plus faciles à gérer cognitivement par l'alternance consonnes-voyelles due audit cadre.

Séquencement rythmique pour la synthèse : les points de contrôle

En s'appuyant sur la théorie frame/content, Delalez enrichit la notion de centre perceptif [6] pour rendre compte de la richesse du rythme, non seulement dans l'organisation des syllabes entre elles, mais aussi au sein d'une syllabe, notamment le rapport de durées entre voyelle et consonne. Pour ce faire, il introduit une nouvelle notion, les *points de contrôle du cadre* (ou FCP, pour *Frame Control Points*).

Comme leur nom l'indique, les FCP sont plus spécifiquement liés au contrôle de la synthèse vocale performative. Ils se répartissent en deux types : les points *nucléiques* et les points de liaison. Ainsi, à chaque syllabe γ, ou plus précisément, à chaque groupe rythmique (correspondant à une ouverture-fermeture de la théorie frame/content) correspondent un point *nucléique* $P_n(\gamma)$, situé au sein du groupe, et un point de *liaison* $P_l(\gamma)$.

Les P_n correspondent ainsi, dans le cadre de la théorie frame/content, à la position ouverte de la mâchoire, et les P_l à la position fermée.

I.3 Vokinesis

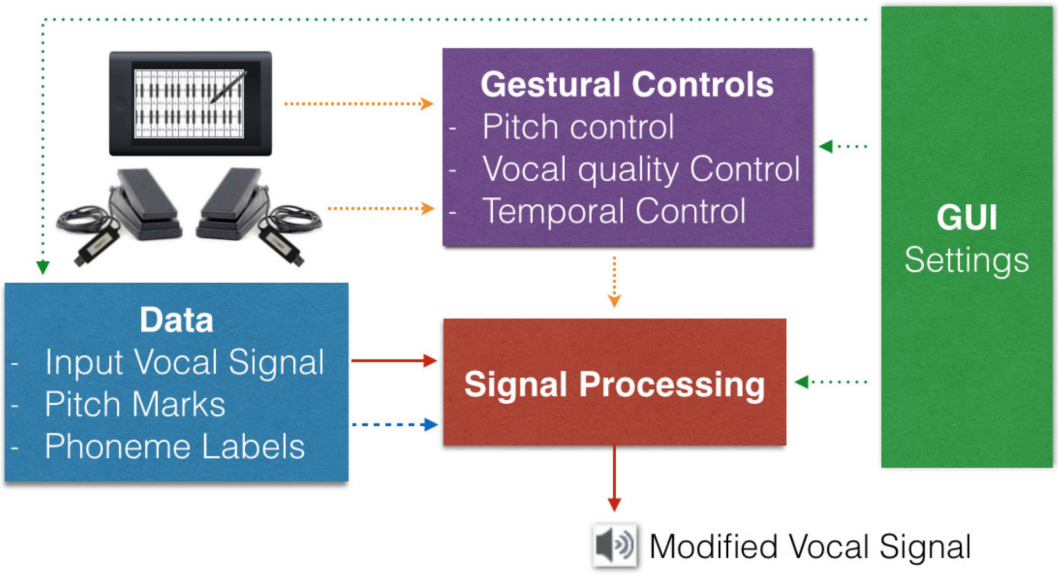


FIGURE I.3 – Vue d'ensemble de l'architecture de Vokinesis, tirée de [5]. Flèches vertes : réglages effectués dans une interface graphique. Oranges : données contrôlées en temps-réel. Bleue : données d'analyse du signal original. Rouges : signaux audio.

Dans le cadre des recherches menées au LAM sur la synthèse vocale performative, plusieurs *instruments chanteurs* [19][13] — des instruments de musique produisant de la voix — ont été développés. Vokinesis [6] est l'un des derniers instruments de cette lignée. L'instrument Voks, développé dans le cadre de ce stage (voir chapitre III), étant très largement inspiré de son fonctionnement, il convient de s'y attarder.

I.3.1 Principe général

Développé par Samuel Delalez dans le cadre de sa thèse [5], Vokinesis, dont le fonctionnement général est schématisé en figure I.3, permet la prononciation de phonèmes arbitraires tout en offrant un contrôle de l'articulation assez fin. Son contrôle s'effectue *via* deux interfaces : une tablette graphique, et une autre interface pour contrôler le rythme, dont le fonctionnement sera détaillé en section I.3.2.

Vokinesis ne crée pas de la voix en synthèse pure : il faut lui fournir en entrée l'enregistrement audio pré-traité d'une voix, qui sera lors de l'utilisation de l'instrument rejouée à un rythme et une mélodie déterminées par l'utilisateur. Outre cet enregistrement, il faut fournir à Vokinesis un fichier spécifiant quels phonèmes il contient, et comment ils sont délimités. Cette délimitation peut être partiellement automatisée par un logiciel spécialisé tel que Praat [2].

Une fois les fichiers nécessaires fournis, Vokinesis détermine la position des points de contrôle (voir section I.2.2) à partir des informations liées aux phonèmes. On peut alors « jouer » la voix initiale en contrôlant la hauteur avec la tablette. La pression du stylet permet également de contrôler l'effort vocal. Le contrôle du rythme peut se faire de plusieurs manières.

I.3.2 Modalités de contrôle du rythme

Dans Vokinesis, le contrôle du rythme peut se faire selon plusieurs modalités. Certaines, de « bas niveau », reproduisent les fonctionnalités de Calliphony [19], l'un des prédécesseurs de Vokinesis ; c'est le cas des modes *scrub* et *speed*. D'autres, de plus haut niveau, font appel aux méta-informations phonétiques et exploitent la notion de points de contrôle. Ces modes, le mode *tap* et le mode *fader*, font la nouveauté de Vokinesis.

Mode *scrub* En mode *scrub* (*frotter* en anglais), outre la dimension horizontale, réservée pour le contrôle de la fréquence, on exploite la dimension verticale de la tablette graphique. Chaque valeur d'ordonnée correspond à un point temporel dans l'enregistrement d'origine. Ainsi, en laissant le stylet au bas de la tablette, on rejouera continûment un son du début de l'enregistrement, et pour jouer l'enregistrement plus ou moins lentement, il faut monter le stylet avec la vitesse voulue.

En mode *scrub*, si l'on met de côté les variations liées à l'effort vocal, chaque point de la tablette correspond à une sonorité spécifique : un timbre, avec une hauteur. Ce n'est pas le cas du mode *speed*.

Mode *speed* En mode *speed*, la dimension verticale de la tablette est aussi exploitée, mais c'est à présent pour contrôler la vitesse de restitution de l'enregistrement initial : si le mode *scrub* permettait de contrôler la *position* dans l'enregistrement original, le mode *speed* permet de contrôler la *dérivée* de cette position.

Ces modes de contrôle sont accompagnés dans Vokinesis de deux modes, nouveaux par rapport à Calliphony.

Mode *tap* Le mode *tap* est la manière la plus simple d'exploiter les points de contrôle : il nécessite d'ajouter une nouvelle interface très simple en plus de la tablette graphique, une touche de clavier alphanumérique. Chaque pression ou relâchement de la touche fait passer d'un point de contrôle au suivant : un pression,

d'un point de de liaison $P_l(\gamma)$ à un point nucléique $P_n(\gamma)$, et un relâchement, d'un point nucléique à un point de liaison. Ainsi, la touche appuyée correspond à une position ouverte du conduit vocal, et la touche relâchée, à une position fermée.

La vitesse de transition d'un point de contrôle au suivant peut être réglée *via* l'interface graphique de Vokinesis. Ce paramètre est délicat à fixer : trop rapide, et la voix n'est pas réaliste, et moins intelligible. Trop lent, et l'instrument prend du retard lors des passages rapides au cours de l'interprétation.

Mode *fader* Le mode *fader* reprend le principe du mode *tap*, mais l'utilisation d'une interface continue, constituée d'une ou deux pédales d'effets, rend le contrôle du rythme plus organique. Là où la pression ou le relâchement d'une touche déclenchaient le passage au point de contrôle suivant avec une vitesse constante, les pédales permettent en mode *fader* de contrôler précisément comment se déroule le passage au point suivant.

Chapitre II

Le vocodeur World

II.1 Fonctionnement de World

World [24] est un vocodeur, c'est-à-dire un logiciel qui comprend un module d'*analyse* de la voix et un module de *synthèse*. S'il ne correspond pas exactement au modèle source-filtre décrit en I.1, il en est largement inspiré. Son fonctionnement est résumé en figure II.3 et détaillé dans cette section.

Le module d'analyse transforme un enregistrement de voix seule, au format audio, en une représentation *paramétrique* du son, qui rend possibles des manipulations telles que :

- modification de la fréquence fondamentale,
- manipulation fine de l'enveloppe spectrale,
- manipulations indépendantes de la partie voisée (voyelles) et de la partie non voisée (consonnes, souffle, chuchotement) du son,
- modifications temporelles : changements de vitesse, lecture à l'envers, bouclage sur un instant du son.

Cette représentation paramétrique n'est utile qu'à condition qu'un module de synthèse permette de la transformer à nouveau en signal audio audible. C'est le cas avec World, qui est en outre utilisable en temps différé comme en temps réel.

II.1.1 Analyse

Paramètres en sortie

Étant donné un enregistrement de voix, on va calculer des paramètres à une série de M instants espacés d'une durée Δ, la période d'analyse, fixée par défaut à 5 ms. On procède à trois étapes, qui permettront d'aboutir à une représentation du son constituée :

- d'une liste $(f_m)_{m\in(1..M)}$ de fréquences, qui correspondent chacune à la fréquence fondamentale pour une trame, estimée à l'aide de l'algorithme Harvest [23].
- d'une liste $(\Gamma_m^i)_{\substack{i\in(1..N_{fft})\\ m\in(1..M)}}$ de tableaux de coefficients, qui correspondent chacun à la densité spectrale de puissance pour une trame, estimée à l'aide de l'algorithme CheapTrick [21] ; pour chaque m, le signal est fenêtré autour de l'instant correspondant $m\Delta$ avec une longueur de $\frac{3}{f_m}$;
- d'une liste $(R_m^i)_{\substack{i\in(1..N_{fft})\\ m\in(1..M)}}$ de tableaux de valeurs comprises entre 0 et 1, qui correspondent au rapport entre partie non voisée et partie voisée du son pour chaque bande de fréquence, estimé à l'aide de l'algorithme D4C [22]. Ici encore, pour chaque m, on utilise une taille de fenêtre proportionnelle à la période fondamentale — à savoir $\frac{4}{f_m}$.

Le procédé est schématisé dans la partie « analyse » en figure II.3. Notons que la connaissance de la fréquence fondamentale est nécessaire pour appliquer les algorithmes CheapTrick et D4C.

Calcul de la fréquence fondamentale

La fréquence fondamentale du signal est calculée en utilisant l'algorithme Harvest [23]. Plusieurs valeurs instantanées sont d'abord calculées tout au long du signal, puis les résultats aux instants voisins sont comparés pour améliorer l'estimation.

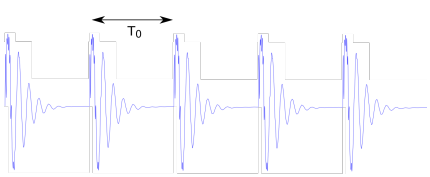
À chaque instant $m\Delta$, une fréquence candidate est d'abord calculée dans chacune de nombreuses bandes de fréquence (40 par octave). Elle est estimée en déterminant l'écart entre les pics et celui entre les zéros de la forme d'onde filtrée. La vraisemblance d'une estimation de fréquence est déterminée en comparant les résultats dans des bandes de fréquence voisines : si une fréquence f est bien la fréquence fondamentale du signal, les valeurs dans les bandes de fréquences proches de f doivent concorder. On obtient un certain nombre de candidats à un certain nombre d'instants espacés de Δ. Les valeurs sont alors raffinées en examinant la fréquence instantanée, définie dans [14].

En s'aidant de la connaissance des valeurs à des instants voisins, les valeurs de fréquence sont raffinées : certains candidats sont éliminés, des valeurs manquantes sont éventuellement ajoutées... Ces opérations permettent de parvenir à un contour de fréquence, sur lequel une dernière interpolation est réalisée.

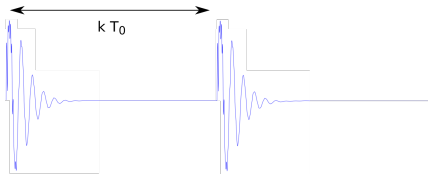
Calcul de l'enveloppe spectrale

L'algorithme CheapTrick [21] permet le calcul d'une enveloppe spectrale. Pour comprendre à quoi correspond cette enveloppe, intéressons-nous à la démarche adoptée

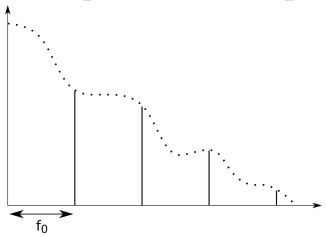
pour changer la fréquence fondamentale d'une voix. Supposons pour simplifier la discussion que le signal n'est constitué que d'une partie périodique.



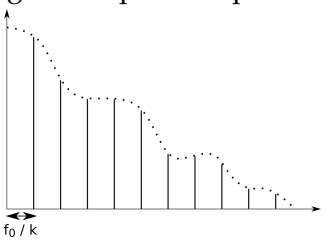
(a) Signal temporel initial, de période T_0



(b) Signal temporel de période kT_0



(c) Spectre du signal initial : fréquence fondamentale f_0



(d) Spectre du signal de fréquence fondamentale $\frac{f_0}{k}$

FIGURE II.1 – Un signal temporel parfaitement périodique de période T_0 (a) présente un spectre constitué de pics régulièrement espacés (c). En déterminant une enveloppe continue (en pointillés) qui fixe la hauteur de ces pics, on peut procéder à un rééchantillonnage dans le domaine spectral (d) et synthétiser un signal équivalent à une autre fréquence (b)

On suppose qu'un signal de voix peut être modélisé par un train d'impulsions, la source glottique, convolué avec la réponse d'un filtre dont les caractéristiques sont liées à la forme du conduit vocal :

$$s(t) = \left(\sum_{n \in \mathbb{Z}} \delta(t - nT_0) \right) * h(t) \text{ — figure II.1 (a)}$$

Dans le domaine spectral (figure II.1 (c)), cette relation s'écrit simplement :

$$S(\omega) = \left(\omega_0 \sum_{n \in \mathbb{Z}} \delta(\omega - n\omega_0) \right) H(\omega) \text{ — figure II.1 (c)}$$

où ω_0 est la pulsation associée à la période T_0, et $H(\omega)$ est la transformée de Fourier de la réponse $h(t)$.

Pour espérer synthétiser un signal similaire, mais en changeant la période T_0, il faut déterminer l'enveloppe spectrale $H(\omega)$ (en pointillés sur la figure II.1 (c) et (d)) ; cette connaissance nous permettra le cas échéant de procéder à un rééchantillonnage dans le domaine spectral :

$$S'(\omega) = \left(\omega_0 \sum_{n\in\mathbb{Z}} \delta(\omega - n\frac{\omega_0}{k})\right) H(\omega) \text{ — figure II.1 (d)}$$

et enfin d'obtenir le signal

$$s'(t) = \left(\sum_{n\in\mathbb{Z}} \delta(t - nkT_0)\right) * h(t) \text{ — figure II.1 (b)}$$

L'enveloppe $H(\omega)$ contient donc des informations utiles à des manipulations telles que le changement de hauteur sans modification des formants. Son calcul s'inspire largement de celui utilisé dans l'un des prédécesseurs de World, TANDEM-Straight [18] ; il comporte plusieurs étapes.

L'opération de changement de hauteur reposant sur un rééchantillonnage dans le domaine spectral, les outils utilisés pour le rééchantillonnage temporel — un problème fondamental en traitement du signal numérique —, rappelés dans [31] et [30], sont dans TANDEM-Straight adaptés au domaine spectral et utilisés. L'analyse cepstrale [3] étant un outil largement utilisé lorsqu'il est question de modèle source-filtre, CheapTrick combine cette approche avec un passage dans le domaine cepstral.

Fenêtrage du signal Le signal fourni est tout d'abord multiplié par le carré d'une fenêtre de Hann, qui a la particularité d'avoir une longueur multiple de la période T_0 du signal — plus précisément, sa longueur est égale à $3T_0$. On peut alors calculer le spectre de puissance du signal ainsi fenêtré, dont on montre qu'il est indépendant de la position temporelle de la fenêtre.

Lissage spectral Ce spectre de puissance obtenu, on souhaite le rendre robuste à une future application du logarithme, qui n'est pas borné pour des valeurs proches de 0. Pour ce faire, on procède à un lissage spectral en lui appliquant un filtre moyenneur, dont la longueur dépend là encore de la fréquence fondamentale du signal, puisqu'elle est égale à deux tiers de cette fréquence :

$$P_s(\omega) = \frac{3}{2\omega_0} \int_{-\frac{\omega_0}{3}}^{\frac{\omega_0}{3}} P(\omega + \lambda) d\lambda \tag{II.1}$$

Séparation source-filtre On peut alors passer dans le domaine des quéfrences par application du logarithme, puis de la transformation de Fourier inverse

$$p_s(\tau) = \mathrm{F}^{-1}\left[\log(P_s(\omega))\right](\tau) \tag{II.2}$$

et effectuer un liftrage en multipliant le cepstre par

- une fonction l_q qui correspond à la troncature aux deux premiers termes de la réponse d'un filtre de correction numérique, comme décrit dans [18]

$$l_q(\tau) = \tilde{q}_0 + 2\tilde{q}_1 \cos\left(\frac{2\pi\tau}{T_0}\right) \tag{II.3}$$

- et un sinus cardinal pour lisser encore une fois le spectre de puissance.

$$p_l(\tau) = p_s(\tau)\,\mathrm{sinc}(\pi f_0 \tau) l_s(f_0) \tag{II.4}$$

Le spectre de puissance obtenu correspondant

$$P_l(\omega) = e^{\mathrm{F}[p_l(\tau)](\omega)} \tag{II.5}$$

correspond alors à celui de l'enveloppe spectrale ; on considère moyennant l'hypothèse évoquée en fin de I.1 qu'il ne contient plus l'information liée à la source. C'est ce spectre de puissance qui sera utilisé dans la synthèse.

Calcul du coefficient apériodique

Le signal de voix, s'il est fondamentalement périodique, contient aussi une partie de bruit, qui est prépondérant dans les phonèmes non voisés, mais intervient également dans les phonèmes voisés, où il correspond alors au souffle. Pour tenir compte de ce phénomène, on calcule pour chaque bande de fréquence un rapport d'apériodicité, une valeur entre 0 et 1 qui indique la quantité de bruit dans le signal dans la bande de fréquence considérée : un rapport de 1 indique que le signal est uniquement composé de bruit ; un rapport de 0 correspond à l'absence de bruit dans la bande de fréquence.

Calcul du retard de groupe La détermination du rapport d'apériodicité, décrit dans [22], repose sur le calcul pour toute fréquence du retard de groupe, défini comme la dérivée fréquentielle de l'argument du spectre du signal considéré. Comme dans CheapTrick, une analyse pitch-synchrone est réalisée avec une fenêtre de longueur égale à quatre fois la période fondamentale, afin que ce retard de groupe ne dépende pas de la position temporelle de la fenêtre, et permet de parvenir à une expression de ce retard $\tau_g(\omega)$.

Lissage spectral et suppression de la composante continue Le graphe du retard de groupe en fonction de la fréquence obtenu, on effectue tour à tour

- un lissage spectral

$$\tau_{gs}(\omega) = \frac{2}{\omega_0} \int_{-\frac{\omega_0}{4}}^{\frac{\omega_0}{4}} \tau_g(\omega + \lambda) d\lambda \tag{II.6}$$

- et une suppression de la partie à basse fréquence de ce graphe

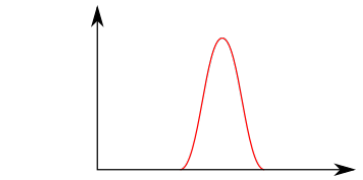
$$\tau_D(\omega) = \tau_{gs}(\omega) - \frac{1}{\omega_0} \int_{-\frac{\omega_0}{2}}^{\frac{\omega_0}{2}} \tau_g(\omega + \lambda) d\lambda \tag{II.7}$$

opérations équivalentes à un filtrage passe-bande *dans le domaine spectral* autour de la pulsation fondamentale ω_0 ; on nomme le paramètre obtenu $\tau_D(\omega)$.

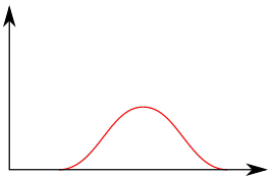
Calcul du rapport d'apériodicité Pour déterminer la quantité de bruit dans une bande de fréquence centrée en ω_c, on prend alors la transformée de Fourier ce paramètre multiplié par une fenêtre de Nuttall w centrée en ω_c.

$$p(t, \omega_c) = \mathrm{F}^{-1} \left[w(\omega) \tau_D(\omega - \omega_c) \right] \tag{II.8}$$

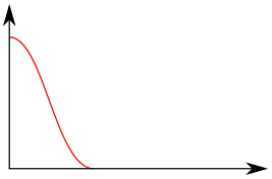
En l'absence de bruit, le signal $p(t, \omega_c)$ obtenu serait un sinus pur de pulsation ω_c ; l'étalement du spectre de puissance de p autour du pic $\omega = \omega_c$ caractérise alors la quantité de bruit dans la bande de fréquence centrée en ω_c.



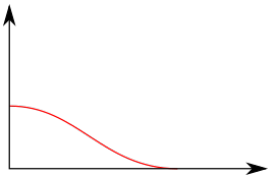
(a) Spectre de puissance concentré



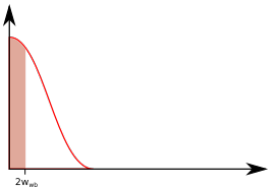
(b) Spectre de puissance étalé



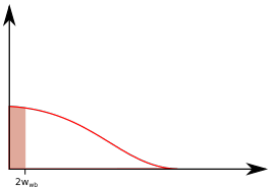
(c) Ordonnancement des valeurs par ordre décroissant



(d) Ordonnancement des valeurs par ordre décroissant



(e) Intégration de 0 à $2w_{bw}$



(f) Intégration de 0 à $2w_{bw}$

FIGURE II.2 – Illustration du calcul du rapport du rapport d'apériodicité pour un p concentré autour d'un pic, et pour un p étalé : un p étalé mène à un rapport d'apériodicité plus élevé.

Pour mesurer cet étalement étant donnée une bande de fréquence centrée en ω_c, le signal p subit plusieurs traitements :

- normalisation, pour que l'énergie totale $\int_t |p(t, \omega_c)|^2$ soit égale à 1,
- réordonnancement des valeurs normalisées $|p_n(t, \omega_c)|^2$ dans l'ordre décroissant (processus illustré en figure II.2 : passage de (a) à (c), et de (b) à (d)), pour s'assurer de bien prendre en compte d'éventuels pics d'énergie secondaires,
- intégration de 0 à $2w_{bw}$, où w_{bw} est une constante qui correspond à la largeur de bande de la fenêtre spectrale w (figure II.2 : (e) et (f)).

Le complément à 1 de la valeur obtenue pour l'intégrale est le rapport r_{ap} recherché. Comme illustré en figure II.2, plus la puissance de p est étalée, plus le rapport d'apériodicité est élevé.

II.1.2 Synthèse

Une fois calculés ces paramètres, on peut resynthétiser le signal d'origine tel quel ou le transformer. Supposons que soient donnés :

- une durée T du signal à synthétiser,
- un profil de fréquence $f_0(t)_{t\in[0,T]}$, qui peut être ou non reliée à la fréquence f du signal d'origine,
- une transformation temporelle qui, au temps de synthèse t, associe le temps d'analyse $\tau = \varphi(t)_{t\in[0,T]}$,

et que l'on veuille synthétiser un signal qui, à chaque instant $t \in [0, T]$, a pour fréquence $f_0(t)$, et pour timbre le timbre du signal d'origine à l'instant $\varphi(t)$.

Notons que des modifications plus subtiles seraient envisageables en manipulant les rapports d'apériodicité et/ou la densité spectrale de puissance. On suppose ici que leur valeur obtenue lors de l'analyse est directement utilisée.

Le fonctionnement qui suit est schématisé dans la partie « synthèse » en figure II.3. Supposons qu'on cherche à reproduire le signal d'origine sans modification, c'est-à-dire que

$$T = M\Delta$$
$$\forall m \in (0..M)\,, f_0(m\Delta) = f_m$$
$$\forall t \in [0, T], \varphi(t) = t$$

La synthèse est *pitch-synchrone* : elle consiste à calculer une série de positions temporelles t_n, espacées de l'inverse de la fréquence fondamentale, et pour chacune d'entre elles, de calculer une réponse $p_n(t)$ correspondant à la partie périodique du signal, et une réponse $a_n(t)$ correspondant à la partie apériodique du signal.

Calcul des instants t_i et interpolation

La première étape consiste à calculer une série d'instants t_n vérifiant pour tout n

$$t_n - t_{n-1} = \frac{1}{f_0(t_n)}$$

Le calcul de ces instants se fait naturellement de proche en proche.

Pour pouvoir par la suite reproduire fidèlement le timbre du signal d'origine au bon moment, on interpole linéairement, pour chacun de ces instants, les valeurs de la densité spectrale de puissance et du rapport d'apériodicité, selon les expressions suivantes :

$$\Gamma^i(t_n) = \lambda \Gamma^i_m + (1-\lambda)\,\Gamma^i_{m+1}$$
$$R^i(t_n) = \lambda N^i_m + (1-\lambda)\,N^i_{m+1}$$

où :

- m est l'indice de trame qui vérifie $m\Delta \leqslant \varphi(t_n) < (m+1)\,\Delta$,
- $\lambda = \frac{(m+1)\Delta - \varphi(t_n)}{\Delta}$.

Calcul de la réponse $p_n(t)$

Étant donné un instant t_n, on calcule d'abord une réponse $p_n(t)$ qui correspond à la partie périodique du signal. Pour ce faire, on commence par calculer une nouvelle enveloppe spectrale, en pondérant pour chaque bande de fréquence l'enveloppe spectrale initiale par le rapport d'apériodicité selon l'expression suivante : $\forall i \in (1..N_{fft})$,

$$\Gamma^i_{per}(t_n) = \big(1 - R^i(t_n)\big) \cdot \Gamma^i(t_n)$$

On calcule ensuite le *spectre à minimum de phase* (MPS sur le schéma en figure II.3) correspondant à Γ^i_{per}, c'est-à-dire le spectre d'un signal dont la densité spectrale de puissance est Γ^i_{per} et dont toute l'énergie est concentrée près de 0. La transformée de Fourier inverse de ce spectre donne la réponse $p_n(t)$ correspondant à la partie périodique du signal.

La fonction $p_n(t)$ est à support fini ; sa taille ne dépasse pas 3 fois la plus grande période de tout l'enregistrement.

Calcul de la réponse $a_n(t)$

On considère toujours un instant t_n ; on calcule à présent la réponse $a_n(t)$ correspondant à la partie apériodique du signal. Le calcul se fait de la même façon, à quelques différences près :

- pondération de l'enveloppe spectrale par le coefficient d'apériodicité (dont le coefficient de périodicité est le complément à 1) :

$$\Gamma^i_{aper}(t_n) = R^i(t_n) \cdot \Gamma^i(t_n)$$

- calcul du spectre à minimum de phase correspondant à la densité spectrale de puissance $\Gamma^i(t_n)$,
- transformée de Fourier inverse,
- convolution de la réponse obtenue avec du bruit blanc.

En pratique, pour alléger le calcul, on ne convolue pas la réponse avec du bruit blanc, mais on réalise l'opération, équivalente, qui consiste à multiplier terme à terme le spectre à minimum de phase correspondant à $\Gamma_{aper}(t_n)$ et le spectre d'un bruit blanc, avant de réaliser la transformée de Fourier inverse.

On obtient la réponse $a_n(t)$ correspondant à la partie apériodique du signal, dont la taille du support est encore une fois inférieur à 3 fois la plus grande période.

Calcul du signal de synthèse $s(t)$

On dispose à présent d'une liste d'instants t_n et de deux listes $p_n(t)$ et $a_n(t)$ de réponses, l'une correspondant à la partie voisée du signal, l'autre à la partie non voisée. Le signal final s'obtient en sommant toutes ces réponses :

$$s(t) = \sum_n (p_n(t - t_n) + a_n(t - t_n))$$

Notons qu'à un instant t donné, plusieurs p_n et a_n sont en général non nuls — on opère un recouvrement des différentes réponses (*overlapp-add*). On sait cependant que leur nombre ne dépassera pas $\frac{3f_0(t)}{f_{min}}$, en vertu des remarques en fin de II.1.2 et II.1.2.

Spécificités du temps réel

En l'état actuel des choses, les valeurs des coefficients spectraux et des rapports d'apériodicité sont directement lues dans les tableaux d'analyse ; l'interpolation décrite en II.1.2 n'a donc pas lieu. Ça n'est pas un problème lorsque l'on synthétise un signal dont le timbre évolue rapidement. En revanche, si, par exemple, on ralentit considérablement la vitesse de lecture d'un signal comportant une consonne plosive (/b/, /p/...), on entend distinctement le passage d'une trame d'analyse à une autre, ce qui n'est pas naturel.

II.2 Aspects pratiques

II.2.1 Limites

World offre une représentation de la voix et un moyen de la resynthétiser qui permettent des applications intéressantes, comme on pourra le voir dans le chapitre suivant, mais il a ses limites.

Temps réel Si la synthèse sonore se fait en temps réel, ce n'est pas le cas de l'analyse. Autrement dit, on peut faire de la synthèse temps réel en s'appuyant sur un fichier audio, mais pas sur un signal lui-même généré en temps réel, comme c'est le cas, par exemple, dans les nombreuses chansons qui font appel à un harmoniseur [1]. Impossible, donc d'appliquer en direct à la voix d'un chanteur les transformations expressives qui seront détaillées dans le chapitre suivant.

Types de sons Outre l'aspect temps réel, World est également limité dans les types de sons qu'il est capable de gérer. Cette limitation n'est pas une surprise, compte tenu de la nature du vocodeur, conçu spécialement pour la voix selon un modèle source-filtre, mais elle rend vains les espoirs de l'utiliser sur n'importe quel autre son pour élargir le champ des possibilités expressives. Plus précisément, World ne permet de gérer que des signaux monophoniques. Des essais montrent ainsi que des chants polyphoniques sont fréquemment remplacés par un simple bruit coloré. Les essais réalisés sur des phrases mélodiques instrumentales sont peu convaincants.

Une exception existe : une simple note d'un instrument peu inharmonique (violon, flûte, ou même piano) est reproduite relativement fidèlement par le vocodeur. Ce n'est en revanche pas le cas d'instruments très inharmoniques (cloche), qui sont, comme le chant polyphonique, remplacés par du bruit.

II.2.2 Exploitation du vocodeur

Dans le chapitre suivant (III), on verra comment World est exploité pour développer un instrument chanteur. Pour utiliser World dans ce cadre, des questions d'ordre légal et technologique se posent :

World est publié sous la licence BSD modifiée, une licence libre permissive. Il nous est donc permis de réutiliser le logiciel comme bon nous semble (modification, diffusion, commercialisation, *etc.*), sous réserve de respecter des règles peu contraignantes (apparition du nom de l'auteur et du texte de la licence — sans pour autant que les produits qui utilisent le code original soient soumis à cette licence). L'aspect légal ne pose donc pas de difficulté.

[1] Wild Love, par Cashmere Cat, The Weeknd et Francis and the Lights, ou encore Hide and Seek (live at House), par Jacob Collier

Le code originel de World est rédigé en C, analyse comme synthèse. Comme on le verra, l'instrument développé au cours de ce stage utilise le langage Max. Les deux opérations se faisant séparément, on fait communiquer ces technologies de deux manières différentes :

Analyse Comme souligné en section II.2.1, l'analyse d'un signal de voix se fait en temps différé. Pas besoin, donc, de faire communiquer directement la partie du programme qui s'occupe de l'analyse et l'instrument chanteur à proprement parler. On choisit donc de réaliser un exécutable externe qui analyse un fichier audio et écrit les résultats de l'analyse dans deux fichiers sur le disque dur. Ces fichiers pourront ensuite être chargés par l'instrument.

Le format des fichiers générés est le format `.jxf.jit`, directement lisible en Max. Utiliser un format directement utilisable par Max nous permettra par la suite d'utiliser des objets (des « fonctions » Max) pour traiter les résultats de l'analyse, plutôt que de recoder par nous-mêmes les fonctions adéquates.

L'interface du programme d'analyse est en ligne de commande et minimaliste : il suffit d'invoquer le nom du programme, le chemin du fichier à analyser et les chemins des deux fichiers à créer/écrire. Le choix d'une interface en ligne de commande peut sembler peu ergonomique. Pour un utilisateur expérimenté, il s'agit pourtant de l'interface la plus simple. Tous les utilisateurs n'ont bien entendu pas d'expérience de la ligne de commande ; cette interface en ligne de commande est cependant la plus simple à envelopper dans une interface graphique dans un éventuel travail postérieur.

Synthèse Au contraire de l'analyse, la synthèse se fait en temps réel, et doit donc être intégrée à Max plus directement. Pour ce faire, on développe un objet Max `jit.world_play` , en utilisant l'interface de programmation fournie par Cycling '74 :

- En entrée, `jit.world_play` prend deux vecteurs, un représentant une enveloppe spectrale et un représentant les rapports d'apériodicité. Il prend également un nombre à virgule flottante qui représente une fréquence fondamentale.
- En sortie, il produit le son correspondant à ces paramètres.

Cette adaptation à Max a d'une part permis de démontrer que le vocodeur est effectivement utilisable en temps réel, et d'autre part de créer l'instrument Voks, décrit dans la partie suivante.

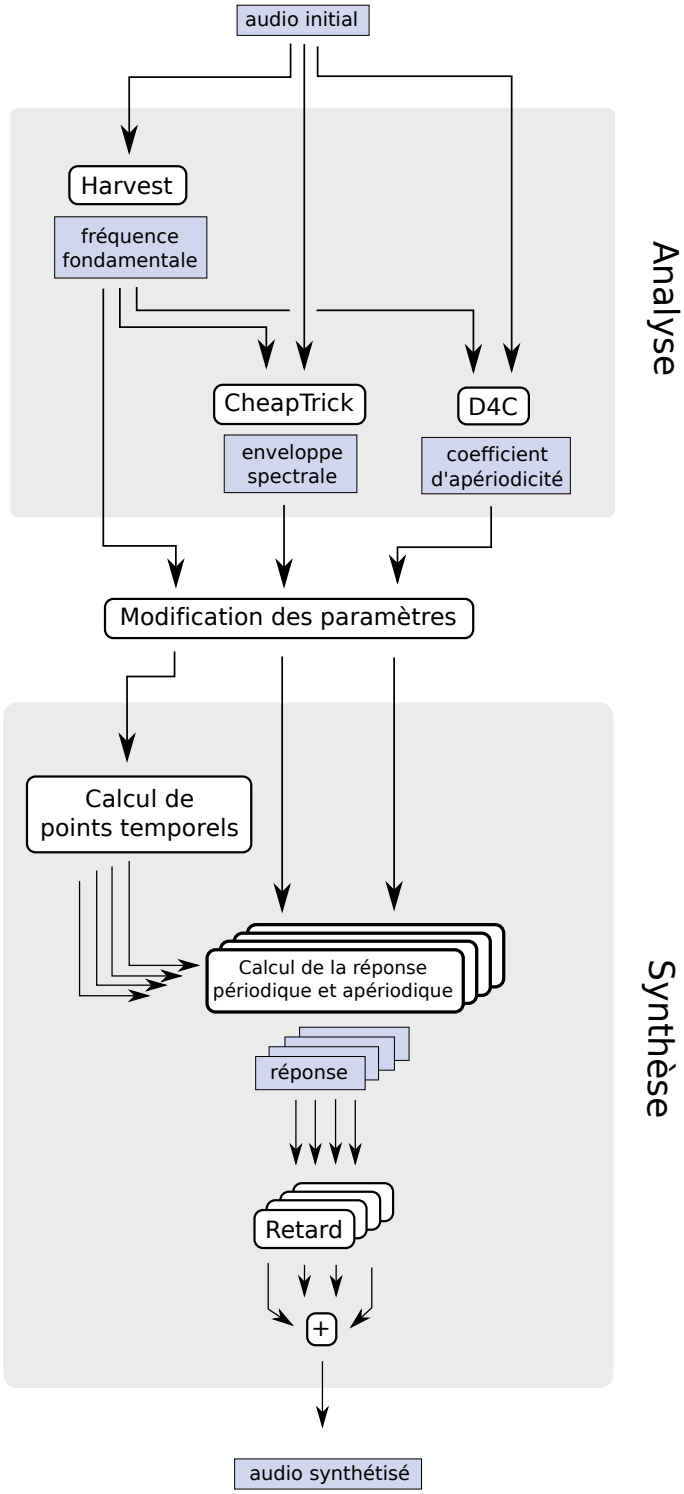


FIGURE II.3 – Schéma de fonctionnement général de World

Chapitre III

L'instrument Voks

Le vocodeur World, dont le fonctionnement a été détaillé en partie II, offre une représentation d'une voix à la fois fidèle — dans la mesure où elle permet de reconstituer le son d'origine de manière indiscernable — et facilement manipulable. En outre, la synthèse est possible en temps réel, comme le prouve notre implémentation sous forme d'un objet Max.

Ces aspects font de World un outil adapté à la synthèse vocale performative. Voks, un nouvel instrument chanteur utilisant World comme moteur audio, a donc été implémenté. Comparable à Vokinesis (section I.3), il permet de jouer une voix en partant d'un enregistrement réalisé au préalable. Il offre sur cette voix un contrôle fin de la prosodie *via* diverses interfaces, mais aussi des modifications expressives de la voix synthétisée qui dépassent les simples variations prosodiques.

Dans cette partie sont détaillés le principe d'utilisation de l'instrument développé, nommé Voks, et sa structure. Pour des informations précises sur l'installation et l'utilisation du logiciel, on peut se référer au fichier `README.markdown` inclus dans le code source de l'instrument.

III.1 Technologies utilisées

Initialement développé par Miller Puckette à l'Institut de Recherche et de Coordination en Acoustique/Musique (IRCAM), aujourd'hui maintenu et commercialisé par la société états-unienne Cycling '74, Max est un langage de programmation graphique qui permet de développer rapidement des applications multimédia temps-réel. Les extensions MSP et Jitter — conçues respectivement pour l'audio, et pour la gestion de matrices — incluses avec Max, permettent d'étendre ses capacités à la musique et au traitement d'images fixes et animées.

Sa licence propriétaire, son peu de portabilité — tout utilisateur d'une application développée avec le langage doit avoir une version de Max (éventuellement gratuite) installée sur sa machine — rendent Max peu adéquat pour le développement de larges applications commerciales. Néanmoins, la rapidité d'implémentation et la

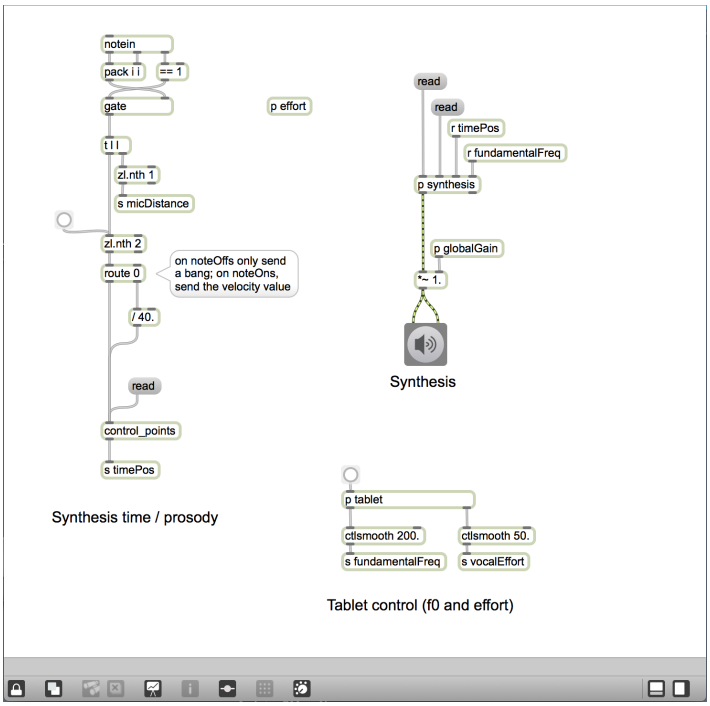


FIGURE III.1 – Capture d'écran du langage graphique Max

possibilité de tester instantanément des « patches » (nom donné aux programmes Max) font de Max un langage pertinent dès lors qu'il s'agit de concevoir un prototype d'instrument tel que celui qui a été développé dans le cadre de ce stage.

III.2 Utilisation de l'instrument

Supposons qu'ait été enregistrée une phrase dans un fichier `voix.wav`, et que l'on souhaite jouer (c'est-à-dire faire parler, ou chanter) cette voix.

- La voix doit d'abord être analysée en utilisant l'algorithme d'analyse de World. L'exécutable en ligne de commande `audio_to_jxf` permet d'appliquer cet algorithme, et créer deux fichiers `specenv.jxf.jit` et `aratio.jxf.jit`, qui contiennent le résultat de l'analyse.
- Il faut ensuite repérer les points de contrôle (cf. section I.2.2) sur l'enregistrement initial, c'est-à-dire reporter dans un fichier texte `ctlpts.txt` les positions temporelles des points de contrôle de la phrase prononcée. Cette tâche, actuellement manuelle (et, reconnaissons-le, un peu fastidieuse) devrait pouvoir être automatisée sans difficulté fondamentale, la délimitation de phonèmes étant un problème déjà résolu par des logiciels tels que Praat [2]. Dans le cas où cette tâche est effectivement automatisée, il est toutefois possible qu'il faille procéder à des ajustements manuels.

- Une fois créés, ces trois fichiers peuvent être chargés dans Voks, et l'on peut jouer la phrase enregistrée, en utilisant deux interfaces de contrôle : une tablette graphique sous une main, et un clavier alphanumérique, un clavier MIDI ou un Touché sous l'autre.

III.2.1 Interfaces de contrôle

Interfaces de contrôle de hauteur et effort

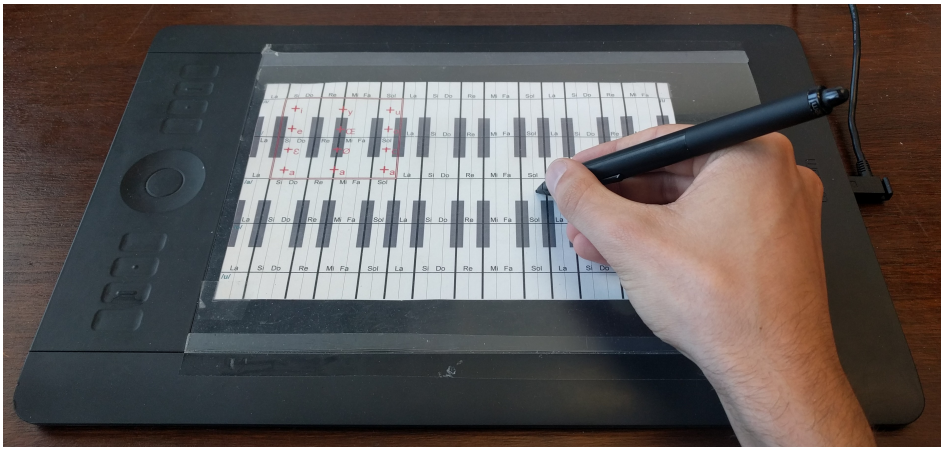


FIGURE III.2 – La tablette graphique, interface de contrôle pour la hauteur et l'effort vocal

Tablette graphique La tablette graphique est utilisée au stylet. Voks exploite deux valeurs : la position de la pointe du stylet selon l'axe horizontal, et la pression appliquée sur cette pointe. La position horizontale du stylet contrôle la fréquence fondamentale du son synthétisé : plus le stylet est situé vers la droite, plus la fréquence fondamentale est élevée, et des écarts de position équivalents correspondent à des intervalles de même valeur. Un guide peut être superposé à la tablette, comme en figure III.2, pour aider au contrôle de la justesse, notamment dans des applications musicales.

La pression appliquée au stylet, quant à elle, est directement liée à l'effort vocal : une pression nulle (stylet relevé) entraîne une effort vocal nul, donc un son inexistant ; une pression maximale (stylet appuyé au maximum) entraîne un effort vocal maximal.

Thérémine Parallèlement à la tablette graphique, la possibilité d'utiliser un thérémine comme interface de contrôle a également été explorée. Inventé par Lev Theremin en 1920, le thérémine est un instrument de musique qui a la particularité d'être joué sans le toucher. Deux antennes permettent respectivement de contrôler la fréquence fondamentale et l'amplitude d'une onde monophonique : plus la main

source : apple.com

source : korg.com

FIGURE III.3 – Trois interfaces de contrôle du rythme possibles : un clavier alphanumérique, un contrôleur MIDI, et le Touché

gauche est proche de l'antenne de gauche, plus le son est faible ; plus la main droite est proche de l'antenne de droite, plus il est aigu.

S'il s'agit d'un instrument analogique, certains thérémines comprennent des sorties de contrôle qui peuvent être traitées numériquement. C'est ainsi qu'alternativement à la tablette graphique a été explorée la possibilité d'utiliser un thérémine pour contrôler la hauteur et l'effort vocal. La sortie qui correspond à la main gauche (habituellement, contrôle du volume) est alors liée à l'entrée d'effort vocal. La sortie qui correspond à la main droite (contrôle de la fréquence fondamentale) est naturellement liée à l'entrée de fréquence fondamentale.

Par rapport à la tablette graphique, le thérémine présente l'avantage d'être un vrai instrument de musique, avec sa technique, son répertoire et ses virtuoses. Certes, l'instrument est moins courant que beaucoup d'autres, mais une théréministe confirmée sera plus à même d'exploiter les possibilités expressives de Voks en utilisant son instrument que quelqu'un qui utilise une tablette graphique, un instrument qui n'a pas d'histoire.

Touche de clavier, capteur de pression, Touché : interfaces de contrôle du rythme

Si l'aspect « contrôle » de la fréquence fondamentale et de l'effort vocal s'avère assez direct, il n'en va pas de même pour ce qui est du contrôle du rythme. Étant donnée une suite de points de contrôle, il faut une interface de contrôle qui permette de spécifier quand passer d'un point de contrôle au suivant, et comment.

Clavier alphanumérique La solution la plus simple consiste en une interface qui permet d'envoyer des signaux ponctuels, comme une simple touche de clavier alphanumérique. Chaque pression ou relâchement de la touche fait passer la lecture au point de contrôle suivant, avec une vitesse v prédéterminée. Cette interface, bien que satisfaisante pour des rythmes lents, présente le problème suivant : la vitesse de jeu d'une phrase est limitée par la vitesse d'enregistrement divisée par v. Si l'on tente de jouer un rythme rapide, le synthétiseur n'aura pas le temps de suivre et sera en retard sur le geste de l'interprète. Une solution serait d'augmenter la vitesse

v, mais si cette dernière est trop élevée, les consonnes passent trop rapidement et le son n'est plus naturel. Une idée est donc d'utiliser une interface de contrôle qui permet de spécifier une vitesse plus grande quand le rythme va devenir rapide.

Capteur de pression Une touche de clavier alphanumérique est une interface courante, mais impossible à utiliser lorsque les deux mains sont sollicitées, comme dans le cas du thérémine. Dans ce cas, l'utilisation d'un *capteur de pression* s'avère appropriée. Il s'agit d'une pastille circulaire, assez petite pour être fixée sur un doigt, qui transmet la valeur de la pression qui lui est appliquée. Pour faire de cet objet une interface de contrôle binaire, comme la touche de clavier alphanumérique, il suffit de fixer une valeur seuil de pression, en-dessous de laquelle la « touche » sera considérée comme appuyée, et au-dessus de laquelle elle sera considérée comme relâchée. Collé au pouce, un tel capteur peut alors être utilisé conjointement au thérémine ; il suffira alors de presser et relever son index pour contrôler le défilement des syllabes.

Contrôleur MIDI classique Pour spécifier une vitesse, le paramètre « vélocité » associé à toute note MIDI semble approprié : il suffit pour y avoir accès de remplacer notre clavier alphanumérique par n'importe quel contrôleur MIDI (clavier MIDI, contrôleur à pads...). Le passage d'un point de contrôle au suivant se fait alors toujours lorsqu'une touche est appuyée, mais la vitesse est à présent d'autant plus grande que la vélocité MIDI associée à la note est élevée, c'est-à-dire que le doigt appuie fort sur la touche. Cette manière de contrôler la vitesse fait l'hypothèse que lorsque l'utilisateur a besoin de jouer rapidement, les pressions de la main qui contrôle le rythme se font plus énergiques.

Touché La vélocité MIDI présente l'avantage d'être disponible sur du matériel très répandu, mais malgré son nom, elle ne correspond pas réellement à une vitesse. Une dernière interface permet, elle, de capter directement la vitesse avec laquelle appuie la main : né au LAM, inspiré par le « retour d'effort » propre à la touche d'effort des Ondes Martenot, à présent fabriqué par l'entreprise française Expressive E, le Touché pourrait être décrit comme une sorte de pédale pour la main très sensible. Il offre quatre dimensions de contrôle, mais une seule nous suffit ; elle correspond à l'enfoncement de l'avant du Touché.

III.3 Structure

L'instrument Voks permet de convertir des données issues des interfaces de contrôle en un signal audio, en s'appuyant sur les informations contenues dans des fichiers externes. La figure III.4 résume la structure de Voks ; dans cette section, on détaille le fonctionnement de ses différents blocs.

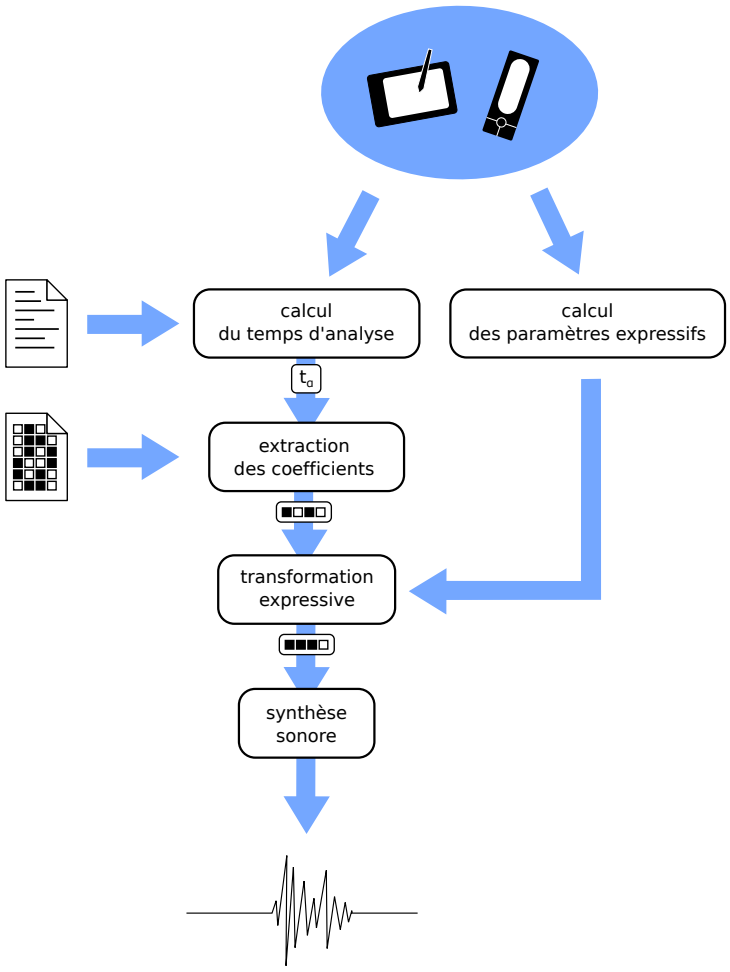


FIGURE III.4 – Structure générale de Voks

III.3.1 Calcul du temps d'analyse

Entrées

- Points de contrôle temporels [6] consignés dans un fichier texte
- Données issues d'un contrôleur

Sortie

- Toutes les cinq millisecondes, un nombre à virgule flottante, correspondant à un instant (en secondes) τ de l'enregistrement initial, le *temps d'ana-*

lyse

La partie *Synthesis time / prosody* permet de charger un fichier texte contenant l'information des points de contrôle p_i correspondant à un fichier audio existant. Elle capte également les messages envoyés par le contrôleur de rythme (section III.2.1). En sortie, elle donne à tout instant (avec un taux de rafraîchissement de 5 millisecondes) une valeur τ, le *temps d'analyse*, qui correspond à l'instant de l'enregistrement audio initial dont il faudra lire les paramètres World.

Pour générer ce temps d'analyse, on utilise les messages reçus par le contrôleur de rythme, qui signalent le moment de passer d'un point de contrôle au suivant. Dans le cas d'une touche de clavier MIDI, il s'agit des *NoteOn* et *NoteOff* ; pour une touche de clavier alphanumérique, des moments d'appui/relâchement de la touche. Pour le Touché, un message est envoyé à chaque fois que la position dépasse un certain seuil, dans un sens, puis dans l'autre.

Une fois qu'un message a été envoyé, à l'instant t_m, le temps d'analyse en sortie $\tau(t)$ passe du point de contrôle courant p_i au suivant p_{i+1}, avec une certaine vitesse v_i :

$$\tau(t > t_m) = \min\left(p_i + v_i(t - t_m), p_{i+1}\right)$$

La manière de déterminer la vitesse v_i dépend de l'interface de contrôle :

- Pour le clavier alphanumérique ou le capteur de pression, v_i est constante et paramétrable. Si l'on ne souhaite pas jouer plus rapidement que le signal d'origine, on peut fixer v_i simplement à 1, pour que les transitions sonnent aussi naturellement que dans le signal en question. Si l'on souhaite jouer plus vite que le signal d'origine, il faudra régler v_i à une valeur plus élevée que 1.

- Avec une interface MIDI, les choses sont plus flexibles : la norme MIDI offre une dimension de contrôle, la *vélocité*, à l'appui d'une touche ou la pression d'un pad. La valeur de cette vélocité est utilisée pour fixer la valeur de v_i jusqu'à l'appui d'une touche suivant : plus la vélocité est élevée, plus la vitesse v_i est rapide. Plus précisément, les valeurs de vélocité variant de 1 à 127, mais une vélocité très faible (moins de 10 ou 5) étant difficile à obtenir, on se contente d'une vitesse v_i proportionnelle à la vélocité.

- Avec le Touché, on capte la valeur de la vitesse, c'est-à-dire la dérivée de la valeur MIDI, à chaque franchissement de seuil. Là encore, la vitesse v_i est proportionnelle à la valeur absolue de cette vitesse.

III.3.2 Extraction des coefficients World

Entrées

- Fichiers World

- Temps d'analyse τ

Sortie

- Coefficients World correspondant à l'instant τ de l'enregistrement original

Le premier bloc de la partie synthèse permet de charger les deux fichiers qui contiennent les coefficients World d'un fichier externe. Une fois chargés ces fichiers, Voks a en mémoire l'information de tout le signal audio, du début à la fin. Le bloc prend également en entrée le temps d'analyse τ. À partir des coefficients de tout le signal, il extrait les coefficients correspondant uniquement à l'instant τ.

III.3.3 Calcul des paramètres expressifs

Entrée

- Valeur de l'effort vocal

Sortie

- Valeurs des paramètres expressifs

Comme détaillé en section III.4, les variations expressives possibles sont diverses (variation de la quantité de bruit, pente spectrale, etc.). Chacune de ces variations fait appel à un paramètre distinct. On n'a pourtant accès qu'à une valeur de contrôle, la pression du stylet sur la tablette. Il faut donc convertir une valeur en un ensemble de valeurs de paramètres, que l'on nommera « paramètres expressifs ». Comme on le verra dans les sections suivantes, ces paramètres visent à contrôler :

- le rapport entre partie périodique et partie apériodique du son (voir la section III.4.1),
- le rapport entre première harmonique et harmoniques supérieures (voir la section III.4.2),
- la forme de la pente spectrale qui sera appliquée (voir la section III.4.3).

Pour gérer les deux premiers éléments de cette liste, on se contentera (sections III.4.1 et III.4.2) de séparer le signal vocal entre une partie apériodique et une partie périodique, elle-même composée de la première harmonique et des harmoniques supérieures. Ces trois parties seront pondérées par des coefficients, qui constituent les trois premiers paramètres expressifs. Le dernier paramètre expressif est la fréquence de coupure de la pente spectrale qui sera appliquée au signal. Finalement, les quatre paramètres sont :

- p_{ap}, le coefficient de pondération de la partie apériodique,

- p_{h_1}, le coefficient de pondération de la première harmonique,
- $p_{h_{sup}}$, le coefficient de pondération des harmoniques supérieures,
- f_c, la fréquence de coupure de la pente spectrale.

L'étape de détermination des paramètres expressifs, qui présente peu de complexité théorique, est délicate et cruciale pour rendre l'instrument convaincant ; il s'agit de doser l'influence du paramètre d'entrée « effort vocal » sur les différents paramètres en sortie, et pour peu que le dosage d'un paramètre soit fait un peu trop à la légère, c'est tout le réalisme du son produit qui disparaît.

Ce dosage doit impérativement être fini à l'oreille pour s'assurer du réalisme du son produit, mais pour le guider, on s'appuie sur [10] et [32], qui donnent les caractéristiques de différents régimes de voix. Les fonctions qui, à une valeur d'effort vocal μ (comprise entre 0 et 1), associent une valeur de p_{ap}, p_{h_1}, $p_{h_{sup}}$ et f_c, obéissent finalement aux contraintes suivantes :

- $p_{ap}(\mu)$, $p_{h_1}(\mu)$, $p_{h_{sup}}(\mu)$ et $f_c(\mu)$ sont continues ;
- $p_{ap}(0) = p_{h_1}(0) = p_{h_{sup}}(0) = 0$; autrement dit, pour une valeur d'effort vocal nulle, aucun son ne sera généré ;
- Pour des valeurs de μ assez faibles, $p_{ap}(\mu)$ est sensiblement supérieur à $p_{h_1}(\mu)$ et $p_{h_{sup}}$, pour produire un chuchotement ;
- Pour des valeurs de μ assez faibles, $p_{h_1}(\mu)$ est sensiblement supérieur à $p_{h_{sup}}$, pour produire un murmure ;
- Pour des valeurs de μ élevées, $p_{ap}(\mu)$, $p_{h_1}(\mu)$ et $p_{h_{sup}}(\mu)$ sont proches de 1, quoique $p_{h_{sup}}(\mu)$ soit supérieur à $p_{h_1}(\mu)$;
- $f_c(\mu)$ est croissante.

Les valeurs en décibels des coefficients de pondération p_{ap}, p_{h_1} et $p_{h_{sup}}$ sont finalement des fonctions de l'effort μ affines par morceaux, choisies pour obéir aux contraintes ci-dessus, à l'exception de l'exigence de nullité, impossible à atteindre avec des valeurs en décibels. Pour satisfaire cette dernière contrainte, on multiplie le tout par une fonction, également continue et affine par morcaux, mais en amplitude, qui vaut 0 pour $\mu = 0$, et 1 pour μ significativement supérieur à 0.

Notons enfin qu'un coefficient multiplicateur de p_{ap} est de surcroît réglable *via* l'interface de Voks, pour un contrôle plus complet du timbre de la voix synthétisée.

III.3.4 Transformation expressive des coefficients World

Entrée

- Coefficients World correspondant à l'instant τ de l'enregistrement original
- Valeurs des paramètres expressifs

Sortie

- Coefficients World modifiés

Une fois obtenues les valeurs des différents paramètres, les variations expressives décrites en section III.4 peuvent être appliquées. C'est à cette étape que la représentation paramétrique de World est exploitée : au lieu de travailler sur un signal audio, on peut manipuler directement les valeurs des coefficients, et ainsi doser exactement la quantité de bruit, isoler les différentes harmoniques, appliquer des filtrages arbitrairement complexes...

Notons que cette représentation est ici convertie dans un format plus facile à manipuler : là où (voir chapitre II) World représente une voix par les coefficients Γ^i — valeur de l'enveloppe spectrale — et R^i — valeur du rapport entre quantité d'apériodicité et quantité de son totale — (dépendants de la bande de fréquence considérée), on préfère travailler directement sur l'enveloppe périodique et l'enveloppe apériodique :

$$\begin{aligned} \Gamma^i_{per} &= \Gamma^i(1 - R^i) \\ \Gamma^i_{aper} &= \Gamma^i R^i \end{aligned} \tag{III.1}$$

Une fois modifiées, ces enveloppes seront reconverties en coefficients utilisables par World :

$$\begin{aligned} \Gamma'^i &= \Gamma'^i_{per} + \Gamma'^i_{aper} \\ R'^i &= \frac{\Gamma'^i_{aper}}{\Gamma'^i_{per} + \Gamma'^i_{aper} + \epsilon} \end{aligned} \tag{III.2}$$

où ϵ est un nombre strictement positif négligeable, introduit pour éviter de risquer une division par 0.

III.3.5 Synthèse sonore

Entrée

- Coefficients World modifiés

Sortie

- Signal audio

La dernière étape est naturellement de convertir les paramètres World auxquels on est parvenu en un signal audio, le tout en temps réel. C'est ici qu'on utilise l'algorithme de synthèse World temps réel, utilisable dans Voks grâce à la création d'un external Max.

III.4 Modifications expressives de la voix synthétisée

Dans cette section, on détaille les étapes de détermination des paramètres expressifs et de modification des coefficients World évoquées en section III.3. Les idées qui motivent ces traitements peuvent être expliquées, mais les réglages précis des paramètres — autrement dit, la mise en correspondance d'un ensemble de valeurs de paramètres pour chaque valeur de l'effort vocal — qui rendent la synthèse véritablement expressive et réaliste ne peuvent être déterminées qu'à l'issue d'un réglage manuel subjectif.

III.4.1 Variation de la quantité d'apériodicité

Lors du chuchotement, seule la partie non voisée d'une voix est produite. En murmurant, on fait entendre la partie voisée, mais la partie non voisée reste prépondérante. Une première manière de faire entendre des variations d'effort vocal est le dosage des parties voisée et non voisée du son. Par chance, World permet de manipuler séparément ces deux parties, moyennant un léger changement de représentation en utilisant l'équation III.1.

Le premier paramètre expressif est donc le rapport global entre la partie voisée et la partie non voisée du son : pour une valeur de l'effort vocal faible, la partie non voisée sera prépondérante, jusqu'à disparaître lorsqu'il est nul. Pour une valeur de l'effort vocal élevé, le rapport entre partie voisée et non voisée est le même que dans l'enregistrement initial. Notons que s'il pourrait être tentant d'aller dans la direction inverse pour un fort effort vocal, et de faire disparaître la partie non voisée, procéder ainsi dénaturerait profondément les consonnes, qui exploitent beaucoup le bruit contenu dans la voix.

III.4.2 Variation de l'équilibre première harmonique / harmoniques supérieures

La représentation de World, si elle est de plus haut niveau que les simples valeurs au cours du temps d'une onde de pression, ne va pas jusqu'à permettre de distinguer et manipuler les formants. Les variations de fréquence et d'amplitude des formants jouent pourtant un rôle dans la perception de l'effort vocal [29], notamment en ce qui concerne le formant glottique [12]. Pour simuler un déplacement du formant glottique, une solution consiste à doser le rapport entre l'amplitude de la première harmonique et les suivantes.

Pour effectuer cette séparation, on travaille sur la partie voisée du son ; en reprenant les notations du chapitre II, il s'agit de

$$\left(\omega_0 \sum_{n\in\mathbb{Z}} \delta(\omega - n\omega_0)\right) H_p(\omega)$$

et le problème revient à changer le rapport entre $H_p(\omega_0)$ et $H_p(|n|\,\omega_0)$, $n > 1$. Pour ce faire, on agit directement sur les coefficients Γ^i_{per}, en séparant la partie Γ^i_{h1} de l'enveloppe périodique qui pondère la première harmonique de la partie $\Gamma^i_{h>1}$ qui pondère les suivantes. On pourrait pour cela se contenter de couper brutalement à $\frac{3}{2}f_0$, entre la première et la deuxième harmonique :

$$\begin{aligned} \Gamma^i_{h1} &= \Gamma^i_{per} \text{ si } iw < \frac{3}{2}f_0, \text{ 0 sinon} \\ \Gamma^i_{h>1} &= \Gamma^i_{per} \text{ si } iw \geqslant \frac{3}{2}f_0, \text{ 0 sinon} \end{aligned} \tag{III.3}$$

où $w = \frac{2F_s}{N_{FFT}}$ est la largeur de la bande de fréquence associée à un i. Il suffit ensuite de pondérer les deux parties de cette enveloppe spectrale pour obtenir les coefficients en sortie :

$$\Gamma'^i_{per} = \alpha\Gamma^i_{h1} + \beta\Gamma^i_{h>1}$$

où α et β sont des paramètres expressifs dépendant de l'effort vocal, à régler manuellement (voir section III.3.3).

En pratique, la grossière discontinuité liée à la coupure à $\frac{3}{2}f_0$ force l'enveloppe spectrale $(\Gamma^i)_{per}$, échantillonnée de manière discrète, à osciller déraisonnablement, entre ces points autour de la discontinuité en $\frac{3}{2}f_0$, allant jusqu'à causer un dépassement du niveau sonore maximal du système et une saturation. On préfère donc utiliser une fonction la plus régulière possible, la tangente hyperbolique. Les équations III.3 sont alors remplacées par

$$\begin{aligned} \Gamma^i_{h1} &= \frac{1}{2}\left(1 + \tanh\left(\frac{iw}{f_0} - \frac{3}{2}\right)\right)\Gamma^i_{per} \\ \Gamma^i_{h>1} &= \frac{1}{2}\left(1 - \tanh\left(\frac{iw}{f_0} - \frac{3}{2}\right)\right)\Gamma^i_{per} \end{aligned} \tag{III.4}$$

III.4.3 Pente spectrale

Comme proposé dans [26], on peut également appliquer une pente spectrale, c'est-à-dire un filtre passe-bas, pour rendre compte d'une baisse d'effort vocal. Là encore, plutôt que de filtrer le signal en sortie, on agit directement sur les coefficients World de la partie voisée, ce qui permet d'appliquer un filtre de réponse en fréquence de

forme arbitraire. On se contente toutefois ici d'appliquer un filtre du premier ordre, dont la fréquence de coupure est un paramètre expressif, dépendant de l'effort vocal, à régler manuellement (voir section III.3.3).

III.4.4 Gain

Les manipulations décrites plus haut changent chacune le niveau sonore global. D'autre part, les variations de ce niveau sonore ont elles-mêmes une influence sur la perception de l'effort vocal [29]. Il est donc nécessaire de l'ajuster, à la fois pour contrebalancer l'effet des autres variations expressives, et comme traitement pour changer directement l'effort vocal. Cette modification est la plus facile à réaliser techniquement, puisqu'il s'agit simplement de multiplier le signal en sortie par un gain, mais comme pour toutes les autres, son paramétrage est délicat.

Sans aucune sorte d'amplification, les variations de niveau sonore d'une voix réelle, notamment lorsque le chuchotement entre en ligne de compte, sont assez importantes pour qu'une chanteuse ou un chanteur utilisant un micro, par exemple, ait besoin de faire varier la distance qui l'en sépare pour ajuster ce niveau sonore. Les variations de niveau sonore entre des efforts vocaux minimal et maximal seront donc souvent plus faibles en présence d'un micro qu'en son absence. C'est ce qu'on pourrait appeler l'« effet micro », que l'usage, courant dans les musiques actuelles, de la compression dynamique, ne fait qu'amplifier. Se pose alors la question de savoir s'il convient de reproduire un niveau sonore comparable à celui d'une voix nue, ou si l'effet micro doit être reproduit.

Pour répondre à cette question, on a ajouté un paramètre qui permet de régler la quantité d'effet micro : quand il est fixé au minimum, le niveau sonore augmente drastiquement entre l'effort vocal faible et celui qui est élevé. Le fixer au maximum, c'est rendre ce niveau sonore plus constant, et laisser le soin aux autres variations (apériodicité, équilibre harmonique, pente spectrale) de rendre compte l'effort vocal.

III.4.5 Limites

Si la représentation paramétrique d'un signal vocal par World est bien de plus haut niveau qu'un simple signal audio, elle ne l'est pas assez pour avoir accès à et modifier des paramètres tels que fréquence et amplitude des formants. Pour des modifications plus intelligentes, des méthodes de détection des formants, telle celle décrite en [28], et de modification pourraient être envisageables, mais on s'est pour le moment cantonné à des modifications globales de l'enveloppe spectrale de la partie périodique.

Enfin, si les modifications proposées ici permettent de diminuer l'effort vocal, elles sont moins à même de l'augmenter :

- Là où une diminution de la partie périodique provoque une impression de

chuchotement, une diminution de la partie apériodique ne ferait que dénaturer le son

- Il n'y a pas d'opération inverse de l'application d'un filtre passe-bas : augmenter artificiellement les coefficients associés aux bandes de fréquence élevée ne fonctionne pas : il faudrait comme dans [26] distordre le signal pour obtenir plus d'harmoniques supérieures, mais cela signifierait modifier directement le signal pendant qu'il est généré, c'est-à-dire modifier le code source de World, ce que l'on s'interdit pour ne pas risquer d'avoir à corriger des bugs difficiles à comprendre.

Chapitre IV

Applications

L'instrument mis au point, il a été utilisé dans différentes situations : d'abord dans une perspective de test et d'évaluation de l'instrument en lui-même, puis pour hiérarchiser les différents paramètres en jeu dans la perception de l'effort vocal. Notons que les utilisations décrites ici, et les constatations qui sont faites, restent informelles. Un travail d'évaluation, avec protocole expérimental perceptif rigoureux, reste à effectuer.

Un exemple d'utilisation de Voks en situation de jeu est disponible à l'adresse suivante : https://peertube.pl/videos/watch/3d6a981e-9e49-462e-ac26-6236d75cc045

IV.1 Élaboration d'exemples sonores

IV.1.1 Enregistrements

Des enregistrements divers ont été utilisés pour évaluer Voks. Si tous ont été réalisés au LAM, ils ont chacun des caractéristiques différentes. Leur point commun est d'être des enregistrements monophoniques de voix en français.

Robert Expert Certains enregistrements sont tirés d'une collaboration de Robert Expert, un contre-ténor qui collabore régulièrement, depuis une vingtaine d'années, avec les chercheurs du LAM. Dans le cadre du développement et de l'utilisation de Vokinesis (voir section I.3), M. Expert a enregistré la traduction française de la *Parabole de la loi* (*Vor dem Gesetz* [17]), un texte extrait du Procès, de Franz Kafka. Le texte a été enregistré deux fois, pendant lesquelles le chanteur s'efforçait de garder une hauteur constante, à environ une octave d'intervalle. Les fréquences fondamentales respectives sont d'environ 190 Hz et 385 Hz.

Ces enregistrements étaient initialement destinés à être joués en concert avec Vokinesis, suivant une partition de Christophe d'Alessandro (disponible en annexe de [5]). Ils présentent l'avantage d'avoir été enregistrés spécifiquement en vue d'un

usage avec un instrument chanteur proche de Voks ; le débit de parole est donc plutôt régulier, et il n'y a pas de variations importantes de fréquence ni d'effort vocal. De surcroît, l'effort vocal de l'enregistrement initial est assez élevé. Comme mentionné en section III.4.5, il est possible de diminuer de beaucoup l'effort vocal, mais presque pas de l'augmenter. Le fait que celui de l'enregistrement initial soit élevé offre donc plus de possibilités expressives.

Le corpus Tatiana D'autres enregistrements sont tirés d'un corpus constitué dans le cadre de travaux sur les variations de qualité des voyelles (fréquences, formants, *etc.*) selon les émotions du locuteur, décrits en [27]. 160 courtes phrases ont donc été enregistrées sept fois par une même locutrice ; six de ces sept enregistrements ont été réalisés avec une émotion spécifique en tête (colère, joie, surprise, *etc.*). Le choix des 160 phrases a été fait de manière à couvrir le plus de groupes de phonèmes de la langue française.

Contrairement aux enregistrement de Robert Expert, ce corpus n'a pas été constitué en vue d'une resynthèse, mais d'une analyse. La hauteur n'y est donc plus constante, le débit, plus rapide, et le timbre variable. Un dernier point rend l'exploitation de l'audio encore plus difficile : certains phonèmes, prononcés rapidement, sont remplacés par d'autres. Le mot [ʃoz], par exemple, est remplacé par [ʃəz] ; cette substitution, acceptable pour une parole rapide, l'est beaucoup moins quand elle est lente.

Enregistrements *ad hoc* En plus des enregistrements évoqués dans les paragraphes précédents ont été réalisés des enregistrements spécialement pour tester Voks. Il s'agit de courtes phrases et d'extraits de chansons célèbres (Edith Piaf, Carmen), et là encore, on s'est efforcé de maintenir une hauteur constante et un effort vocal élevé. Ces enregistrements permettent d'une part de montrer les possibilités de l'instrument, en en faisant la démonstration sur des airs connus de tous, et d'autre part de pouvoir tester l'influence des divers paramètres en jeu dans l'effort vocal.

IV.1.2 Étiquetage

Une fois sélectionnés les enregistrements à exploiter, il a fallu les étiqueter, c'est-à-dire marquer la position des points de contrôle. Cette tâche a été faite manuellement ; comme préconisé dans [5], les points de contrôle nucléiques ont été placés vers les deux tiers de la voyelle, et les points de contrôle de liaison dans la phase qui précède la liaison pour les plosives, et au cœur de la consonne pour les autres. Les étiquetages étant manuels, on ne s'est bien entendu pas interdit de revenir dessus pour obtenir des résultats jugés meilleurs.

IV.1.3 Jeu

Les enregistrements étiquetés, intervient la phase de jeu. Les interfaces utilisées sont principalement les plus simples à mettre en place : tablette graphique et touche de clavier alphanumérique. Différents types de jeu ont été testés (voix parlée, voix chantée, effort vocal très variable ou plutôt fixe). Dans chaque cas, l'audio en sortie est évidemment enregistré ; on enregistre également la valeur des données de contrôle de la tablette (position, pression du stylet) et celle du temps d'analyse τ, pour pouvoir les réutiliser.

IV.1.4 Génération d'exemples sonores avec variation des paramètres

Lors du jeu de l'instrument, tous les éléments qui participent à la modification de l'effort vocal — variation de la quantité d'apériodicité, pente spectrale... — sont présents. Pour mieux comprendre l'influence de ces divers éléments, il est pourtant souhaitable d'en isoler certains. Pour ce faire, les données de contrôle captées en IV.1.3 sont réutilisées pour synthétiser la même phrase plusieurs fois, mais en modifiant la manière dont le son est généré. Pour chaque phrase, jouée avec un effort vocal successivement faible, puis élevé, on génère ainsi plusieurs versions alternatives. Une liste des différentes modifications effectuées, ainsi que des commentaires sur les résultats obtenus, est dressée en annexe IV.A de ce chapitre.

IV.2 Résultats

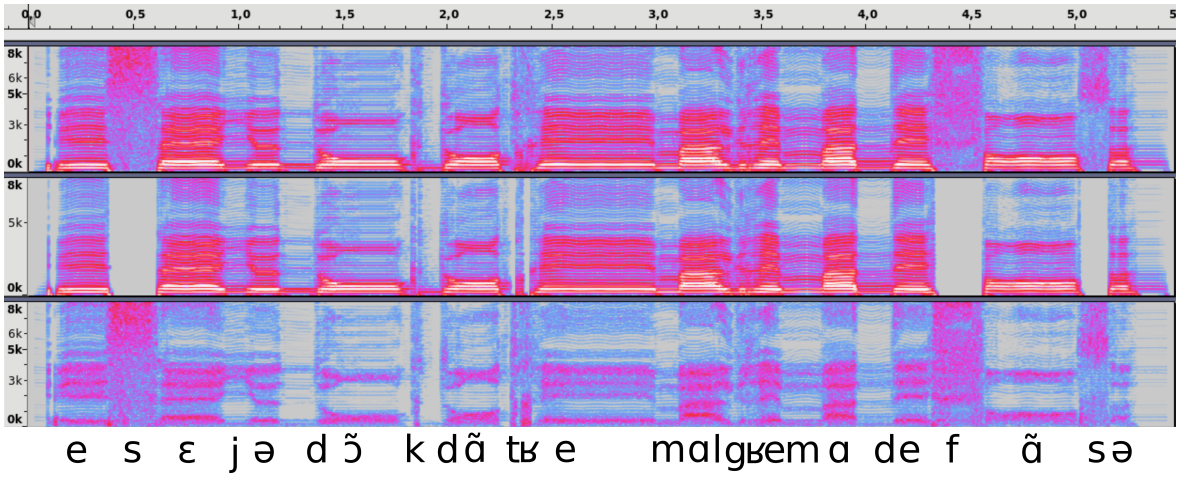


FIGURE IV.1 – Spectrogramme de trois versions de la même phrase *Essaye donc d'entrer malgré ma défense*, générée avec le même geste : en haut, la version entendue lors du jeu de l'instrument, au milieu, avec l'apériodicité réduite à zéro, et en bas, avec la périodicité réduite à zéro.

Une évaluation informelle des exemples sonores produits montre la capacité de l'instrument à faire varier la perception de l'effort vocal : la simulation du déplacement du formant glottique *via* la modification du rapport entre première harmonique et harmoniques supérieures, et surtout la variation de la quantité d'apériodicité permettent de mieux rendre compte de l'effort vocal que la seule pente spectrale, déjà implémentée dans des systèmes tels que Vokinesis.

Les différents exemples réalisés en réutilisant les données de contrôle permettent en outre d'obtenir une gamme d'exemples sonores réalistes qui, s'ils proviennent de données de contrôle identiques, sont clairement distincts. Ainsi, la variation du coefficient multiplicateur de la partie apériodique peut évoquer une voix plus ou moins vieille ou malade ; en fixant la fréquence de coupure de la pente spectrale à son minimum, on peut donner l'impression d'une articulation forte combinée à un voisement faible.

D'autres exemples ne sont pas réalistes, comme lorsque la fréquence fondamentale est fixe, mais certains permettent d'observer des phénomènes intéressants : l'amplification de la partie apériodique combinée à la désactivation de la pente spectrale conduit à une voix assez bruitée pour sembler avoir été murmurée, mais la partie voisée est trop brillante pour que ce soit le cas.

Des commentaires plus précis sur chacun des exemples obtenus sont disponibles en annexe IV.A de ce chapitre.

IV.A Liste des modifications de paramètres effectuées

Cette liste répertorie toutes les modifications effectuées grâce aux modifications des paramètres de Voks et décrit les résultats obtenus.

- **Version originale**

Paramètre modifié : aucun : le son est celui qui a été entendu durant le jeu de l'instrument, avec des paramètres acceptables pour un jeu réaliste.

- **Apériodicité maximale**

Paramètre modifié : le coefficient multiplicateur de l'apériodicité, fixé au maximum accessible *via* l'interface de Voks.

Commentaires : Résultat plutôt réaliste. L'apériodicité étant élevée, on entend beaucoup de souffle dans la voix : pour un effort vocal faible, cet effet fait se rapprocher la voix d'un chuchotement. Pour un effort élevé, effet d'accentuation des consonnes.

• **Apériodicité minimale**

Paramètre modifié : le coefficient multiplicateur de l'apériodicité, fixé au minimum accessible *via* l'interface de Voks.

Commentaires : Résultat moins réaliste. L'énergie de la partie périodique étant plus distribuée sur les hautes fréquences que celle de la partie apériodique, cette modification donne un résultat comparable à l'application d'un passe-bas.

• **Apériodicité nulle**

Paramètre modifié : la quantité de partie apériodique, fixée à zéro.

Commentaires : Résultat non réaliste. Effet de passe-bas accentué, et disparition de la plupart des consonnes, clairement visible sur le deuxième spectrogramme en figure IV.1. La perte d'information est particulièrement importante pour un effort vocal faible, où le spectre de la partie périodique est moins riche.

• **Partie périodique nulle**

Paramètre modifié : la quantité de partie périodique, fixée à zéro.

Commentaires : Effet de chuchotement. Résultat réaliste, mais peu de possibilités expressives puisque, seul le chuchotement étant possible, les variations d'effort vocal n'ont pas d'influence, et la fréquence fondamentale est inexistante.

• **Pente spectrale maximale**

Paramètre modifié : la fréquence de coupure de la pente spectrale, fixée à une valeur minimale, 100 Hz, choisie inférieure à la fréquence fondamentale.

Commentaires : Impression d'une voix souffrante par contraste avec la partie apériodique, non affectée par la pente spectrale. Effet particulièrement important lorsque l'effort vocal est élevé.

• **Pente spectrale désactivée**

Paramètre modifié : la fréquence de coupure de la pente spectrale, fixée à une valeur infinie (en pratique, la partie « pente spectrale » du logiciel est simplement désactivée).

Commentaires : Résultat non réaliste pour un effort vocal faible : la voix paraît trop puissante pour avoir une telle quantité de souffle. Pour un effort vocal élevé, le résultat est presque inchangé.

• **Fréquence constante**

Paramètre modifié : la fréquence fondamentale, fixée à une valeur comparable aux valeurs de jeu, mais constante.

Commentaires : Impression d'une voix robotique. Effet accentué pour un effort vocal élevé, où les variations de fréquence fondamentale sont habituellement plus importantes.

Conclusion

Travail réalisé

Durant ce stage, le vocodeur World, pour lequel n'existait qu'une documentation sommaire, a été étudié et compris en détail. Sa modification a permis de montrer qu'il était possible de l'utiliser en utilisant des données de contrôle en temps réel.

Dans la lignée des instruments chanteurs développés au LAM, l'instrument développé pendant ces quelques mois, Voks, exploite la souplesse de la représentation paramétrique de World pour permettre de jouer une voix en temps réel en introduisant des variations expressives riches. Ce jeu se fait *via* l'interface privilégiée des instruments chanteurs, la tablette graphique, mais peut aussi en utiliser d'autres, moins courantes, pour multiplier les possibilités expressives.

Travail restant, perspectives

Un travail important reste à réaliser ; il s'agit bien entendu de la mise en place d'un protocole d'évaluation perceptive de Voks rigoureux, d'une part pour évaluer l'instrument lui-même, et d'autre part pour évaluer l'influence des différents éléments de l'effort vocal, et le réalisme de leur combinaison.

Le réglage des paramètres de l'instrument effectué en III.3.3 pourrait également être perfectionné, en mettant en place, ici aussi, un protocole d'évaluation perceptive qui permettrait de déterminer quelle valeur des paramètres doit être affectée à chaque valeur de l'effort vocal.

Enfin, d'autres modifications expressives peuvent être explorées : en jouant sur la représentation de World de manière nouvelle, on peut envisager, par exemple, de faire parler ou chanter une voix dont le genre est réglable, de très masculin à très féminin. On peut également envisager des modifications du style de chant : chant lyrique, soul, *etc.*

Bibliographie

[1] John W. Black. Relationships among fundamental frequency, vocal sound pressure, and rate of speaking. *Language and Speech*, 4(4) :196–199, 1961.

[2] Paul Boersma and David Weenink. Praat : doing phonetics by computer [programme informatique], 2018.

[3] Donald G. Childers, David P. Skinner, and Robert C. Kemerait. The cepstrum : A guide to processing. *Proceedings of the IEEE*, 65(10) :1428–1443, Oct 1977.

[4] Christophe d'Alessandro and Boris Doval. Voice quality modification for emotional speech synthesis. In *EUROSPEECH-2003*, pages 1653–1656, 2003.

[5] Samuel Delalez. *Vokinesis : instrument de contrôle suprasegmental de la synthèse vocale*. PhD thesis, Université Paris-Sud, 11 2017. non publiée.

[6] Samuel Delalez and Christophe d'Alessandro. Vokinesis : Syllabic control points for performative singing synthesis. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 198–203, Copenhagen, Denmark, 2017. Aalborg University Copenhagen.

[7] David Deterding. The formants of monophthong vowels in standard southern british english pronunciation. *Journal of the International Phonetic Association*, 27(1-2) :47–55, 1997.

[8] Boris Doval, Christophe d'Alessandro, and Nathalie Henrich. The spectrum of glottal flow models. *Acta acustica united with acustica*, 92(6) :1026–1046, 2006.

[9] Homer Dudley. The carrier nature of speech. *Bell System Technical Journal*, 19(4) :495–515, 1940.

[10] Christophe d'Alessandro. Voice source parameters and prosodic analysis. *Methods in empirical prosody research*, 3 :63–87, 2006.

[11] Gunnar Fant. *Acoustic Theory of Speech Production : With Calculations Based on X-Ray Studies of Russian Articulations*. Number 2. Walter de Gruyter, 1970.

[12] Gunnar Fant. Glottal source and excitation analysis. *STL-QPSR*, 1(1979) :85–107, 1979.

[13] Lionel Feugère, Christophe d'Alessandro, Boris Doval, and Olivier Perrotin. Cantor digitalis : chironomic parametric synthesis of singing. *EURASIP Journal on Audio, Speech, and Music Processing*, 2017(1) :2, Jan 2017.

[14] James L. Flanagan and Richard M. Golden. Phase vocoder. *Bell System Technical Journal*, 45(9) :1493–1509, 1966.

[15] Hiroya Fujisaki and Takako Kawashima. The roles of pitch and higher formants in the perception of vowels. *IEEE transactions on audio and electroacoustics*, 16(1) :73–77, March 1968.

[16] Jessica E. Huber, Elaine T. Stathopoulos, Gina M. Curione, Theresa A. Ash, and Kenneth Johnson. Formants of children, women, and men : The effects of vocal intensity variation. *The Journal of the Acoustical Society of America*, 106(3) :1532–1542, 1999.

[17] Franz Kafka. *Vor dem Gesetz*. 1915. Traduction : Christophe d'Alessandro.

[18] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno. Tandem-straight : A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, f0, and aperiodicity estimation. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3933–3936, March 2008.

[19] Sylvain le Beux, Christophe d'Alessandro, Albert Rilliard, and Boris Doval. Calliphony : a system for real-time gestural modification of intonation and rhythm. In *International Conference on Speech Prosody (SP 2010)*, Chicago, USA, 2010. Paper 101.

[20] Peter F. MacNeilage. The frame/content theory of evolution of speech production. *Behavioral and brain sciences*, 21(4) :499–511, 1998.

[21] Masanori Morise. Cheaptrick, a spectral envelope estimator for high-quality speech synthesis. *Speech Communication*, 67 :1 – 7, 2015.

[22] Masanori Morise. D4c, a band-aperiodicity estimator for high-quality speech synthesis. *Speech Communication*, 84 :57 – 65, 2016.

[23] Masanori Morise. Harvest : A high-performance fundamental frequency estimator from speech signals. In *Interspeech 2017*, pages 2321–2325, 2017.

[24] Masanori Morise, Fumiya Yokomori, and Kenji Ozawa. World : A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems*, E99.D(7) :1877–1884, 2016.

[25] John Morton, Steve Marcus, and Clive Frankish. Perceptual centers (p-centers). *Psychological Review*, 83(5) :405–408, 1976.

[26] Olivier Perrotin and Christophe d'Alessandro. Vocal effort modification for singing synthesis. In *Interspeech 2016*, pages 1235–1239, 2016.

[27] Albert Rilliard, Christophe d'Alessandro, and Marc Evrard. Paradigmatic variation of vowels in expressive speech : Acoustic description and dimensional analysis. *The Journal of the Acoustical Society of America*, 143(1) :109–122, 2018.

[28] Ronald W. Schafer and Lawrence R. Rabiner. System for automatic formant analysis of voiced speech. *The Journal of the Acoustical Society of America*, 47(2B) :634–648, 1970.

[29] Hartmut Traunmüller and Anders Eriksson. Acoustic effects of variation in vocal effort by men, women, and children. *The Journal of the Acoustical Society of America*, 107(6) :3438–3451, 2000.

[30] Michael Unser. Sampling — 50 years after shannon. *Proceedings of the IEEE*, 88(4) :569–587, April 2000.

[31] Michael Unser and Akram Aldroubi. A general sampling theory for nonideal acquisition devices. *IEEE Transactions on Signal Processing*, 42(11) :2915–2925, Nov 1994.

[32] Chi Zhang and John H. L. Hansen. Analysis and classification of speech mode : whispered through shouted. In *Interspeech 2007*, pages 2289–2292, 2007.