Beller Grégory IRCAM - PARIS VIII

Synthèse concaténative de la parole par sélection d'unités

Laboratoire Analyse-Synthèse sous la direction de **Xavier Rodet**

ircam Centre Pompidou

1 place Igor-Stravinsky
75004 Paris
tél. 01 44 78 48 43
fax 01 44 78 15 40

Plan:

Introduction:

La synthèse de la parole par sélection d'unités est aujourd'hui la manière la plus efficace de synthétiser la parole. En effet, l'intelligibilité du résultat confère pour l'instant à cette méthode, un intérêt plus grand que les synthèses basées sur des modèles (modèles physiques paramétriques) ou celles à base de transformations. Aussi désire t-on maintenant ajouter à la voix ainsi générée, un aspect naturel. Cet enjeu passe par la génération de la prosodie.

I] Contexte du stage:

1) de CATERPILLAR à TALKAPILLAR:

La synthèse musicale par sélection d'unités consiste à choisir dans une large base de données les unités sonores les plus appropriées pour construire, par concaténation et modification, la phrase musicale à produire. La thèse de D. Schwarz sur ce sujet s'est terminée en 2004.

Elle présente:

- Constitution d'une large base de données par alignement de partitions.
- Création d'un système de gestion et de sélection: CATERPILLAR
- Applications musicales.

Conséquemment à ses travaux, une application en voix parlée a été aussi envisagée dans le cadre d'un projet de reconstitution de la voix d'un locuteur disparu. Ce stage s'inscrit dans ce projet de synthèse de la parole de haute qualité: TALKAPILLAR. Ce projet vise à synthétiser la voix de locuteurs spécifiques (Jean Cocteau et Gille Deleuze) pour rendre audibles des textes jamais prononcés par ces locuteurs.

Il a donc une double vocation:

-artistique, tout d'abord

-scientifique, dans la mesure ou il participe aux recherches effectuées dans ce domaine.

La génération de la prosodie par sélection d'unités est à mi-chemin entre la synthèse de la parole et la synthèse musicale. D'ailleurs, on parle également de prosodie instrumentale lorsqu'on veut décrire des nuances de pitch expressives (vibrato, pitch bend...) ou des variations de durée propres à l'interprétation d'un instrumentiste.

2) synthèse concaténative par sélection d'unités:

Dans un système de synthèse par sélection d'unités, des segments audio de tailles variables sont sélectionnés dans un grand corpus de parole puis concaténés pour synthétiser un signal de parole extrêmement naturel. La première étape indispensable est l'indexation et la segmentation de la source. La deuxième étape consiste en l'évaluation du meilleur candidat correspondant le mieux possible avec la cible.

On distingue actuellement deux tendances pour le système de sélection:

La première, issue des travaux de Black, Hunt, et Campbell (1996) et utilisée principalement par AT&T (US) et ATR (Japan), procède par minimisation dynamique d'une fonction de coût, estimée à partir de la phrase à produire (et de ses caractéristiques linguistiques) et des phrases enregistrées dans une base de données (ces phrases étant elles-mêmes analysées en fonction des mêmes critères linguistiques que la phrase à produire). La base de données n'est pas organisée de façon particulière. Les unités disponibles ne sont pas regroupées en fonction de leurs similitudes spectrales. Cette approche, utilisée par Diemo Schwarz est la base de CATERPILLAR.

La seconde, qui résulte d'une thèse de doctorat déposée par Robert Ed. Donovan à Cambridge en 1996, organise au contraire la base de données de façon à pouvoir choisir rapidement l'unité requise, à partir de ses critères linguistiques. Le plus souvent, il s'agit d'une classification en arbre, effectuée une fois pour toutes, lors de la conception du synthétiseur. La taille de l'arbre est représentative de la finesse de la modélisation et peut donc être adaptée à l'inventaire des segments disponible. La sélection d'unités ne se fait qu'entre classes dont les contextes sont adéquats par opposition à une sélection globale.

Enfin, la dernière étape est la synthèse par concaténation des unités sélectionnées. Les algorithmes de concaténation sont conçus pour modifier les segments sélectionnés (par des transformations de base sur la hauteur et la durée des unités) et les concaténer de façon que les discontinuités (énergie, F0, formants, qualité de la source...) au point de concaténation soient réduites avec le moins d'artefacts possibles pour ne pas dégrader le naturel des segments de départ. Une des dimensions principales qui influe sur la dégradation audio est la distance entre la courbe de F0 originale et la courbe cible. L'algorithme TD-PSOLA (IRCAM) qui est utilisée dans CATERPILLAR possède cette propriété intéressante que si le mouvement est nul, la dégradation de qualité est nulle, ce qui n'est pas le cas de l'algorithme MBROLA (Dutoit 1996) issu du laboratoire de la faculté polytechnique de Mons, qui introduit une dégradation constante quel que soit la modification de F0. Nous utilisons donc TD-PSOLA dans notre système TALKAPILLAR.

II] La prosodie:

1) Définition:

La structure prosodique résulte d'interactions complexes entre différents niveaux d'organisation sémantico-pragmatiques, syntaxique et rythmique. Elle se manifeste par le jeu simultané de plusieurs paramètres acoustiques: la fréquence fondamentale Fo, le timbre, l'intensité, la durée des phonèmes. Perceptivement, la hauteur et son évolution, le rythme et le tempo (débit), le registre et le timbre mais aussi les pauses et les silences nous permettent la compréhension d'informations au-delà des mots prononcés. C'est cette deuxième partie du double codage de la parole qui lui confère un caractère "naturel" et évite la monotonie. Elle permet entre autre de véhiculer des informations ectolinguistiques ou phonostylistiques (expressivité, sentiments), de lever des ambiguïtés de sens entre deux phrases phonétiquement similaires et de structurer l'énoncé.

La variation de hauteur est certainement l'indice acoustique le plus important dans la prosodie. Le registre couvert par la plupart des locuteurs est souvent divisible en 4 niveaux perceptivement distinguables: Nous les nommerons:

H+H+ : niveau le plus haut

HH

LL

L-L- : niveau le plus bas

La fréquence fondamentale Fo évolue dans ce registre. Son évolution au cours du temps décrit des contours . Une phrase est généralement composée d'une suite de contours qui ne suivent pas nécessairement la même orientation de pente. On observe cependant une déclinaison générale qui correspond à un abaissemnt de Fo du début à la fin de l'énoncé. La hauteur la plus basse correspond donc à la fin de cet énoncé et constitue ainsi un bon indice de segementation. Ce phénomène à priori universel est de nature physiologique, mais il est géré par le locuteur à des fins linguistiques; il permet de délimiter la fin d'une phrase syntaxique. Il faut remarquer que l'on ne peut évaluer cette fréquence fondamentale que sur les segments voisés (voyelles et quelques consonnes...). Aussi, nous extrapolons celle-ci durant les segments non voisées afin d'avoir des contours continus.

2)La prosodie du Français:

Le français est une langue à accent fixe ou accent de groupe (de mot). Elle se distingue ainsi des langues à accent libre comme l'anglais. L'anglais est une langue très musicale, caractérisée par de fortes variations de hauteurs et couvrant une large tessiture. Il utilise principalement les variations de hauteur et d'intensité. Les tons mélodiques sont très difficiles à acquérir pour les Français dont la tessiture est restreinte. D'autre part, l'organisation rythmique de l'anglais est complètement différente de celle du français. L'anglais est une langue stress timed (Pike, 1947) où l'accent n'est pas prédictible, mais l'espace entre deux pics accentuels est à peu près stable. A l'inverse, la place de l'accent tonique en français est totalement prédictible puisqu'elle affecte toujours la dernière syllabe du groupe rythmique.

On distingue deux types d'accents qui mettent en relief la phrase:

-L'accent primaire (ou tonique) se traduit par un allongement de durée et une variation significative de Fo. Il a une fonction structurante et peut se déduire de la syntaxe.

-L'accent secondaire se manifeste par des variations plus subtiles de Fo et de l'intensité. Il a une fonction focalisante, rhétorique ou expressive.

Cette distinction est fondamentale car elle met en valeur la différence fonctionnelle de ces deux accents. Ils sont les marqueurs temporels et acoustiques de deux types de groupes:

-Les groupes intonatifs (qui se terminent par un accent primaire). Ils expriment la modalité de la phrase. Ils ne sont pas congruents à la syntaxe mais la syntaxe est congruente à l'intonation.

-Les groupes accentuels (qui contiennent un accent secondaire). Ils mettent en relief des mots.

Les groupes intonatifs comprennent généralement un ou plusieurs groupes accentuels. Mais cet imbriquement et cette différence de durée n'impliquent en rien une hiérarchisation entre ces deux éléments car ils ne possèdent pas la même fonction:

-L'intonation:

expression de la structure, modalité

=>invariance monotonie

expression normalisée

=>cadre normatif

adhésion à une structure sociale

force de cohésion

globale (phrase)

-L’accent:

expression individualisée

=>variabilité,subjectivité

excursion mélodique

=>expression originale

rupture fondatrice d’une individualité

force de dissociation

locale (mot)

L’intonation permet de manifester la modalité de la phrase en français:

-phrase assertive: contour descendant du niveau haut au niveau moyen

h

m

b

-phrase impérative: contour descendant linéairement du niveau haut au niveau bas

h

m

b

-question partielle ou interrogation: contour courbe descendant du niveau haut au niveau bas

h

m

b

-question totale: contour courbe montant du niveau bas au niveau haut

h

m

b

Le laboratoire de morphosyntaxe de Paris III (1991-1997) propose d'affiner cette description par contours en positionnant le locuteur sur ce qu'il dit: Chaque niveau de hauteur étant le reflet de ce positionnement:

-H+H+: Mise en place de la co-énonciation

-HH : Consensualité acquise

-LL : Niveau neutre

-L-L-: Rupture de la co-énonciation/égocentrage

L'attitude monologale (contour descendant du LL au L-L-) et l'attitude dialogale (contour montant du HH au H+H+) deviennent des descripteurs de contour efficients pour exprimer les modalités suivantes:

-l'incise (parenthèse): accélération du débit ( )

-la négation: discordance de point de vue

désengagement du locuteur

-la question: valeur neutre et consensuelle ? => .

changement de thème, demande de confirmation ? => .

suscite une réaction, énonciative ?

-l'exclamation: appel à une convergence de point de vue !

ironie, égocentrage suivant un consensus !

surprise, discontinuité dans le fonctionnement de la pensée !

3) les modèles accentuels de la phrase française:

Les recherches sur la prosodie ont aboutit à de nombreux modèles accentuels de la phrase française:

-Ph. Martin (1980) construit des règles fondées sur les corrélations entre relations syntactico-sémantiques et le caractère accentuelle des unités. Des relations de dépendances syntaxiques (simple et à distance) construites "linéairement" (de gauche à droite) permettent de discerner l'unité accentuable.

-S.P. Verluyten (1982) établit une hiérachisation de la phrase en groupes intonatifs (déduits de la syntaxe) et en groupes accentuels (déduits de la longueur des mots) et ajoute à ces règles lexicales des règles de réajustement rythmique pour tenir compte du principe d'isochronie de la langue française (égalité de durée entre deux syllabes non marquées (NA)).

-F. Dell (1984) prédit le degré accentuel grâce à un réajustement rythmique par degré d'eurythmie (qui n'authorise pas que deux accents forts se suivent par exemple).

-Une extension du modèle de Ph. Martin (1986-1987) s'affranchit de la structure syntaxique. Elle propose plusieurs découpages en groupes accentuels et choisit celui ayant la meilleure eurythmie.

-A. Di Cristo et D. Hirst (1993-1996) construisent une grille métrique grâce à des règles eurythmiques permettant l'attribution des accents. Leur démarche diffère dans le sens ou ils définissent le rythme comme l'évolution simultannée de la durée et du ton.

-Pasdeloup (1990) prétend que l'accent est un processus de groupement rythmique. Il instaure une hiérarchie en quatre niveaux allant de la phrase à la syllabe. Quatre règles génèrent en suite la prosodie.

-E. Delais-Roussarie (1995) utilise la théorie de l'optimalité: La génération accentuelle s'effectue en trois étapes:

-Génération de candidats (Groupes intonatifs et groupes accentuels)

-On fait passer ces candidats dans trois modules parallèles de contraintes hierarchisées (modules syntaxe, rythme et sémantique)

-On évalue le candidat optimal selon une hiérarchie de contraintes.

Tous ces modèles sont issus d'observations et aboutissent pour la plupart à des jeux de règles. Qu'ils partent d'analyses syntaxique, phonologique ou rythmique (psycho-acoustique), ils permettent de mieux comprendre d'ou proviennent les paramètres acoustiques de la prosodie.

Cependant, il convient de se demander s'ils sont adaptés à la prédiction prosodique pour une génération automatique qui se veut naturelle et surtout personnalisée. Peux t'on envisager une construction de la prosodie par règles dans le cadre de notre mission artistique? L'élaboration de tous ces modèles visent à obtenir une vision globale et généraliste de la structuration prosodique. Dans tous les cas, ces modèles ont été élaborés dans l'optique de prédire l'évolution des paramètres acoustiques de n'importe quel locuteur. Cela revient à dire que, par conception, ces règles ne peuvent aboutir qu'au caractère normalisé de notre expression.

En effet, de nombreux modèles ne cherchent à prédire que l'apparition des accents primaires, qui sont les indices de la modalité (frontières des groupes intonatifs). Elles ne mènent que rarement aux marqueurs accentuels (accents secondaires) propres à l'expressivité et dont les apparitions révèlent la "personnalité prosodique" de chacun. Une approche par règles nous est donc prohibée si nous voulons restituer dans des phrases synthétisées, la personnalité d'un locuteur spécifique. En l'occurence, il se trouve que Gilles Deleuze est particulièrement expressif de part son intonation.

4) La prosodie dans la synthèse de la parole:

La synthèse de la prosodie apparait clairement indispensable pour tout système TTS (Text To Speech) qui désire véhiculer des informations que ne peut contenir les mots seulement. On distingue dans la littérature, trois méthodes pour la génération de la prosodie:

-L'approche par règles

-L'approche basée sur l'apprentissage à partir de corpus:

*par réseaux de neuronnes

*par HMM (Hidden Markov Models)

*par d'autres méthodes statistiques...

-L'approche par sélection d'unités.

La connaissance de patrons intonatifs ou contours types permet aux domaines de la reconnaissance et de la synthèse de la parole d'élaborer des modèles de l'intonation française:

-Au CNET (1977-1989): On étudie un corpus pour en extraire un jeu de règles qui attribue un patron intonatif en fonction de la syntaxe.

-Chez IBM (1971-1980): On construit un jeu de règles statuant 9 contours types selon le nombre de syllabes, le nombre de mots... On distingue quatre niveaux dans une phrase: phrase, proposition, groupe et mot. L'auteur précise que les niveaux phrase et groupe suffisent pour la majorité des énoncés. Cela revient un peu à négliger les accents secondaires.

-G. Bailly (Grenoble) (1983): Il segmente aussi la phrase en groupes de respiration, de phonation, de sens. Leurs tailles est généralement comprises entre 8 et 12 syllabes.

Pour genérer les contours, il utilise le modèle de H. Fujisaki.

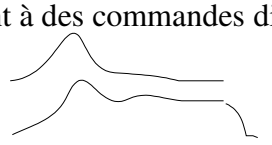
---

le modèle de H. Fujisaki:

La continuité des contours est de nature physiologique. Ils répondent à des commandes discrètes:

-commande de groupe: réponse d'un 2nd ordre à un Dirac.

-commande d'accent: réponse d'un 2nd ordre à un Echelon.



Ce second ordre modélise le muscle crico-thyroïdien (en translation et en rotation). L'avantage de cette modélisation et qu'elle présente des coefficients constants fittables pour chaque locuteur). Seuls l'amplitude et le temps de déclenchement varient. Les trois commandes de groupe sont: Initialisation, réinitialisation, Finalisation. Ils correspondent à l'expression de la modalité.

-V.Aubergé (Grenoble) (1991-1997): Création d'un lexique de contours. Il part de l'autonomie entre syntaxe et prosodie. Grâce à un réseau de neuronnes entrainé sur un corpus, il crée un lexique faisant le lien entre syntaxe et contours prototypiques. C'est aujourd'hui le modèle le plus abouti.

-F.Beaugendre (LIMSI) (1994): Reconnaissance de contours perceptivement pertinents. 30 règles pour la génération de mouvements standards.

Dans le contexte d'une synthèse par concaténation d'unités (allant du semi-phone au mot ou plus), il semble "logique" de sélectionner aussi des unités prosodiques... Mais ce choix vient en fait de motivations plus profondes. En effet, cette approche permet tout d'abord une plus grande variété prosodique que les approches par règles. De plus elle permet de refléter le "caractère prosodique" de l'individu (chacun ayant ses modes d'intonation, registre...), ce qui est essentiel compte tenu de notre but artistique. Enfin, l'introduction de contours réels de Fo sur des blocs de parole permet de conserver la structure micro-mélodique.

5) Automatic prosody generation using suprasegmental Unit Selection (Malfrère, Dutoit et Mertens) 1998:

Le système de l'université polytechnique de Mons que nous allons décrire repose sur la sélection d'unités prosodiques. Il utlise le générateur LIPSS du projet EULER qui génére une description symbolique de la prosodie à partir d'un texte (fichier .txt.mlc):

-une étude syntaxique donne les accents finaux qui délimitent les unités:

-NA: syllabe non accentuée

-AF: syllabe accentuée (accent final=accent primaire)

-UNDEFINED: pause (silencieuse)

-une étude de la modalité donne la hauteur du ton final:

-déclaration: L-L-

-interrogation: HH

-exclamation: H+H+

-temps de pause: P1 ou P2

Ce générateur est appliqué aux phrases de la source comme à celles de la cible. Il permet de créer des unités descripteurs prosodiques de longueurs variables et dont les frontières sont les accents finaux. Ainsi, chaque unité descripteur prosodique possède une clé propre représentant:

-L'index de l'unité dans la phrase

-les tons des accents finaux de début (qui appartient à l'unité précédente) et de fin d'unité

-le nombre de syllabes neutres, inaccentuées dans l'unité

Cette clé peut ressembler par exemple à : "2 FA1NA1NA2NA3FA2 " ou FA1 et FA2 prennent leur valeurs dans {HH, H/H, L-L-, H+H+,N} dans lequel N représente le début d'une phrase. On ajoute aux clés des unités de la source des marqueurs en liens avec le fichier audio aligné qui nous permettent de retrouver les paramètres acoustiques comme l'évolution réelle de Fo durant l'unité... Le choix de l'unité optimale s'effectue en minimisant une fonction de coût. Comme pour le choix d'unités segmentales, cette fonction de coût résulte de l'addition de deux coûts:

-<u>coût de distance à la cible:</u>

-les tones des premier et dernier accents doivent correspondre

-une pondération est ajustée en tenant compte du nombre de syllabes

-une autre est fonction de la position de l'unité dans la phrase

On obtient ainsi une présélection de plusieurs unités candidates.

-<u>coût de concaténation:</u>

Il est seulement basé sur la proximité des valeurs moyennes de Fo de deux unités consécutives. On about it grâce à un algoritme de Viterbi à la sélection finale des unités en choisissant celles dont l'enchaînement présente le coût le plus faible. Puis on va extraire des unités suprasegmentales de la source choisies, les paramètres acoustiques (l'évolution de Fo). Ensuite, on les fournit à l'organe de synthèse (MBROLA) pour que celui-ci appliques des transformations élémentaires à l'enchaînement des unités segmentales choisies en parallèle. Ainsi la phrase synthétisée présente une courbe intonative semblable à celle qu'aurait pu produire le locuteur lui-même.

Enfin le rythme est généré par règles grâce à CART, module du système FESTIVAL (système américain).

6)Remarques individuelles:

Dans la mesure ou nous déployons un système qui cherche à tout sélectionner, aussi bien au niveau segmental que supra-segmental, il me paraît dommage d'utiliser encore des règles pour construire le rythme. Aussi choisissons nous de ne pas rectifier les durées issues directement des unités segmentales.

Nous pourrions aller plus loin dans cette idée de proscription totale de règles, en banissant l'étape fournit par EULER qui visent à déduire de la syntaxe une description symbolique de la prosodie. Il est certain que si cette étape donne une mauvaise description, alors on choisira de mauvaises unités. On pourrait envisagé de construire une fonction de coût ne dépendant que des syntaxes des unités source et cible.

Cependant, dans la mesure ou cette description ne dépend que du lexique et de la modalité, on peut soupçonner qu'elle traduira bien le caractère normalisé de la prosodie, c'est à dire l'aspect conventionné de notre expression. Comme nous l'avons vu, on est en droit de modéliser par des règles l'apparition de groupes intonatifs car la place des accents finaux est systématique. Par contre, on peut difficilement modéliser l'expressivité issues des accents secondaires et de fluctuations plus fines et individuelles. Et c'est en celà que l'approche par sélection d'unités est intéressante. Elle utilise comme descripteurs, les traits communs de tous, pour donner accès aux variations intimes de chacun. Nous choisissons donc de garder EULER, ceci afin de faciliter aussi la description des unités suprasegmentales.

Dans le cadre de CATERPILLAR, de nombreux descripteurs bas niveaux ont été créés et peuvent se révéler très intéressants pour l'évaluation du coût de concaténation: Par exemple la concavité des courbes de Fo...

Le fait que l'on ait la liberté de forcer l'apparition d'une unité plutôt qu'une autre est très importante dans notre optique artistique. Celà nous permettra de choisir perceptivement les contours les plus vraiscemblables.

Enfin, je pense qu'il serait intéressant d'effectuer la sélection des unités prosodiques avant de sélectionner les unités segmentales. En effet, une fois un contour choisi, on peut affecter aux unités segmentales appartenant à ce contour, des poids plus faibles que les autres de manière à favoriser leur apparition. Plus largement, il sera préférable de choisir des unités segmentales dont le Fo sera proche de celui fourni par l'unité prosodique retenue, ceci afin de minimiser la transformation effectuée par l'algorithme TD-PSOLA.

III] Système mis en place: TALKAPILLAR

Le cadre étant défini, rentrons dans les détails... TALKAPILLAR est l'adaptation de CATERPILLAR aux signaux de parole. Nous utilisons donc tout l'environnement crée par Diemo Schwarz comprenant la gestion d'une base de données relationnelle PostgreSQL (Dbi), Les algorithmes de sélection (Viterbi...)... Pour en savoir plus : voir la thèse de Diemo Schwarz.

Il ne nous reste plus qu'à créer des unités prosodiques. Puis il faut adapter ou créer des descripteurs adaptés aux signaux de parole et à la prosodie. Et enfin, il faut régler leurs poids dans la fonction de coût pour la sélection.

La base de données relationnelle nous permet de créer des relations de congruence entre les unités. Nous allons donc crée les unités prosodiques à partir des unités déjà dans la base, grâce aux descripteurs accents et tons, comme dans le système de Dutoit, Malfrère et Mertens.

1) Création d'un corpus:

De manière à avoir dans notre base de données des « instantiations » de tous les diphonèmes possibles de la langue française, cie tous les diphones dont on peut avoir besoin, il faut compléter le texte de l'académie de Cocteau. On y extrait les diphonèmes manquant grâce à un script PERL d'Orsten Karki.

Une fois les diphonèmes manquants (en XSAMPA) extraits, il nous faut les convertir en API, un autre code phonétique. Malheureusement, le passage d'un code à l'autre n'est pas bijectif et il manque entre autre à API, les différenciations entre le « o » ouvert et le « o » fermé. Pourquoi cette transcription ? Car une fois les diphonèmes ou digrammes manquants traduits en API, on va pouvoir chercher dans une grande base de donnée (Lemmes.txt) des mots contenant ces digrammes. Cette base de donnée se présente sous la forme d'un tableau regroupant 50000 mots (lemmes car verbes à l'infinitif), leurs transcriptions phonétiques ainsi que leurs fréquences d'apparition dans un grand corpus de texte et sur le Web (http://www.lexique.org). A chacun des digrammes manquants, on associe le mot de la langue française le contenant et le plus couramment utilisé. Cette sélection s'opère grâce au script PERL: **select_words.pl**. Une fois ces mots sélectionnés, nous extrayons d'un grand corpus de textes (corpatext.txt), des phrases contenant ces mots grâce au script PERL: **select_sentences.pl.**

Récapitulons: pour chaque digramme manquant, nous choisissons le mot le plus couramment utilisé et dont le code phonétique contient notre digramme. Puis nous choisissons une phrase contenant ce mot.

Ainsi, nous avons constitué un texte dans lequel résident tous les cas figures de la prononciation française (élargie par des mots comme camping...).

Ce texte est lu par Xavier Rodet dans la chambre anécoique de l'IRCAM. Le choix de ce « comédien » est du à plusieurs raisons: L'élargissement de la base s'effectuera dans le futur sans le

problème de retrouver un acteur lambda. L'évaluation du résultat de la synthèse se fera sans problème puisque le locuteur « disparu » sera toujours là pour pouvoir comparer. Enfin, la démonstration de notre système se fera certainement par lui et cela permettra donc aux sujets intéressés de comparer directement les résultat.

Cet enregistrement en qualité CD sur DAT est ensuite importé dans la base une méthode décrite ci-dessous.

Il comprend: - Le discours d'entrée à l'académie française écrit par Cocteau.

- Les phrases indépendantes citées ci-dessus.

- L'ensemble des diphones prononcés séparément de manière à faciliter la segmentation.

On aurait pu le faire avec les diphones de Cocteau que Orsten Karki avait segmenté à la main, mais dans un souci de robustesse (et de justesse des marqueurs), nous avons fait le choix de les enregistrer avec la voix de Xavier Rodet. Ainsi la comparaison des diphones prononcés par XR avec le texte prononcé aussi par XR semble plus judicieuse que la comparaison des diphones prononcés par Cocteau avec le texte prononcé par XR.

2) Processus pour créer des unités:

Ci dessous, je décrit la marche à suivre par étapes pour importer des unités dans la base.

2.1) Structure des dossiers et des fichiers:

La hiérarchie des fichiers doit être très claire puisqu'on importe de nombreux fichiers apportant chacun des informations sur les unités. D'ailleurs, de manière à les indexer, tous les fichiers correspondant à un même fichier .aiff (et pas .aif) comportent dans leurs noms un chiffre qui les relient.

Par exemple, CocteauParXavierTrie.536.aiff va de pair avec CocteauXavierSdif.536.sdif. Qu'importent en définitive les noms (bien qu'il est plus simple d'avoir toujours le même référençant le corpus utilisé), tant que l'on respecte les contraintes suivantes:

- Tous les fichiers dont les informations décrivent la même séquence temporelle doivent comporter dans leur nom le même nombre. (ex:536)

- Les noms de ces fichiers ne doivent comporter qu'un seul nombre.

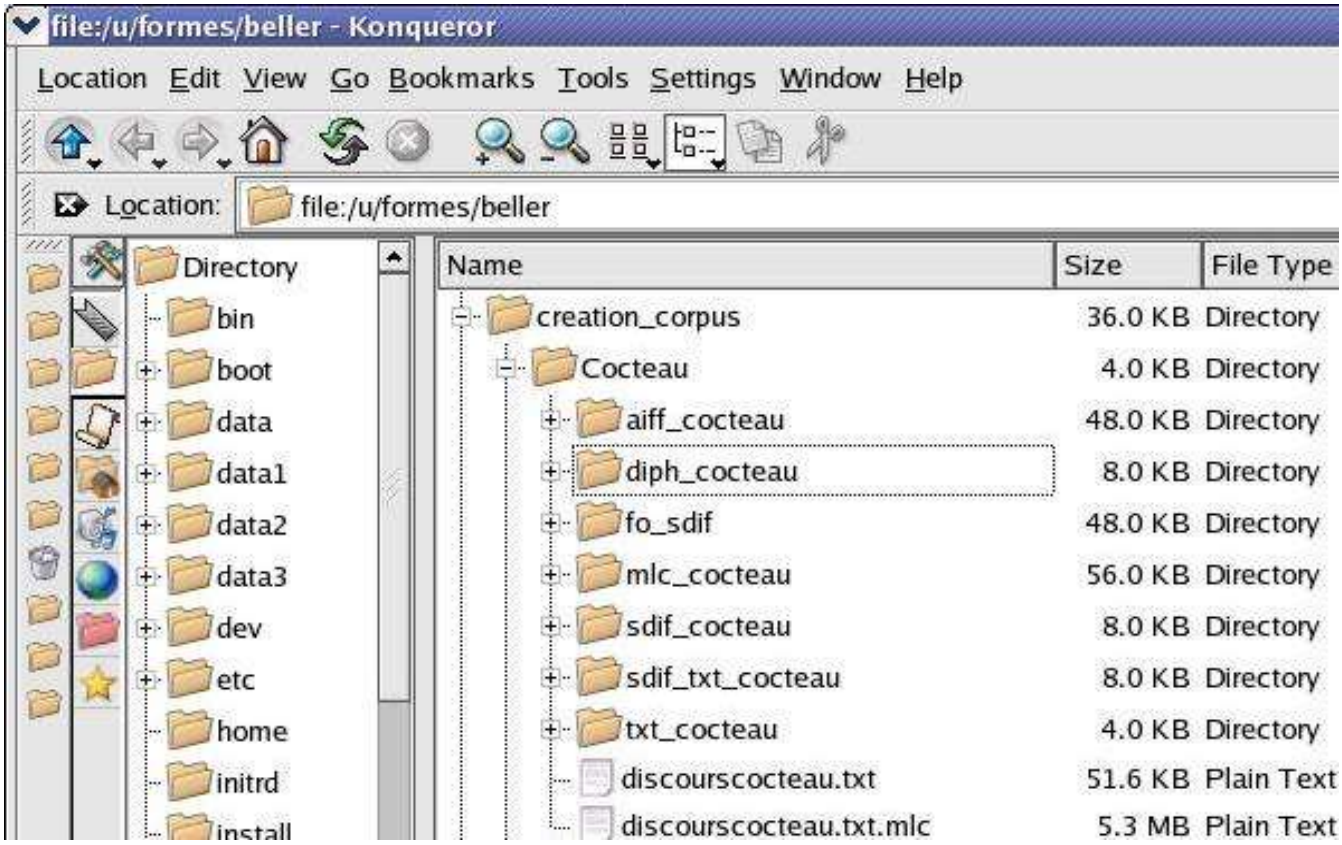
(pas de 5.1.1.1.5.CocteauDiph.536.diph par exemple)

- Une fois dans la base, les noms des fichiers sont dépourvus de leur extension (.aiff, .sdif...). De manière à les différencier, il est bon de changer le nom lorsqu'on change de type de fichiers. Par exemple : CocteauParXavierTrie et CocteauXavierSdif.

Ces règles sont importantes dans la mesure ou l'importation des unités fait appel à plusieurs fichiers et que ceux-ci sont appelés par la fonction MATLAB: **give_the_name_with_the_number.m**

Si des fichiers ne sont pas bons et que l'on a besoin de réindexer tous les autres (si on supprime les fichiers avec le nombre 34, par exemple), on peut utiliser **ReIndexFile.m** qui permet de ne pas avoir de « trous » dans les index.

Pour la gestion des dossiers, on a choisit la structure suivante: (pour le corpus Cocteau, par exemple):



On débute avec seulement deux de ces dossiers:

- aiff_cocteau qui contient les fichiers .aiff issus des enregistrements.

Le meilleur compromis entre taille de fichier dans la base et facilité d'indexation consiste à segmenter les enregistrements en phrases. Chaque phrases étant indexée par un nombre (536 par exemple).

- txt_cocteau qui contient les fichiers .txt correspondant à chaque fichier .aiff.

Les autres dossiers sont issus des différentes analyses réalisées soit sur les fichiers .aiff, soit sur les fichiers .txt.

2.1.1) Analyses des fichiers audio:

2.1.1.1) fo_sdif:

Ce dossier contient les analyses de la fréquence fondamentale des fichiers .aiff réalisée grâce à ADDITIVE.

Pour le générer, un script UNIX à était écrit: **additive_fo.exe** (pas d'arguments)

Il suffit de se placer dans aiff_cocteau et de le lancer via une console.

Les fichiers générés sont des fichiers SDIF que l'on peut visualiser avec **sdiftotext.**

Rq: ce script crée le dossier fo-sdif, puis lance additive pour chaque fichiers .aiff. ADDITIVE crée des dossiers ADDXXXX que l'on supprime après avoir déplacer les fichier .f0.sdif. Enfin il renomme ces derniers en fichiers .fo.sdif de manière à ce qu'il ne comporte qu'un nombre dans leurs noms (0 -> o).

2.1.1.2) diph_cocteau:

Ce dossier contient les fichiers de segmentation, cie de marqueurs temporels. Issus de l'alignement automatique ou bien manuel, ils doivent impérativement comprendre l'extension .diph. Si ce n'est pas le cas (l'alignement génère des fichiers .mx.d), il faut utiliser **renom.exe,** petit script permettant de renommer des fichiers.

Attention, les données de la segmentation doivent impérativement être organisées chronologiquement. Chaque diphone doit se situer dans le fichier .diph par rapport à l'ordre chronologique.

2.1.2) Analyses des fichiers textes:

2.1.2.1) mlc_cocteau:

Ce dossier contient les fichiers .txt.mlc issu de l'analyse d'EULER qui se trouve sur la station WINDOWS de l'équipe. D'ailleurs il serait bon de porter ce module sous UNIX. Le code source est disponible.

on peut accéder à WINDOWS sous UNIX via VNCVIEWER: pour cela, il faut se logger sous SARON:

> ssh saron -X

> password: XXXXXX

> vncviewer pc-bellany

> password: anasyn

Une fenêtre s'ouvre avec l'environnement WINDOWS et EULER se trouve dans ProgramFiles. Pour accéder à vos fichiers .txt, il faut se logger sous WINDOWS à KETHUK avec leechftp.

On rentre le discours en format texte à Euler. Il faut mettre dans les « settings » d'Euler : -phoFileOut suivi du nom du fichier avec l'extension précisée .txt.mlc.

Pour ce que nous voulons faire (traduction texte-phonemes, MLC), il n'est pas nécessaire de synthétiser une voix. Euler peut-être paramétré à l'aide du fichier Euler.ini. Pour l'utilisation du fichier Euler.ini, voir le rapport d'Orsten Karki.

Étant donné que les analyses se font sous WINDOWS avec EULER via VNCVIEWER, il est difficile d'automatiser le tout. Sachant que l'on a un nombre assez conséquent de fichiers .txt (facilement 500 phrases), il paraît plus judicieux de tous les faire en une « passe ».

Ainsi, on concatène préalablement toutes les phrases dans un seul fichier texte (discourscocteau.txt). Chaque phrase doit se terminer par un point (. ! ?) puis un retour chariot. Il ne faut pas sauter de lignes entre les phrases. (utiliser **enleve_retchar.pl** pour les ôter)

En réalité, ce fichier contenant tout le texte existe au départ, et c'est de lui qu'on extirpe les fichiers .txt correspondant à chaque phrases.

Une fois le fichier .txt passé par EULER, on a un fichier .txt.mlc qui doit contenir autant de mlc que de phrases. (il reste souvent une mlc vide à la fin en plus qui correspond au dernier retour chariot du fichier .txt. Si c'est la cas, il faut la supprimer).

Puis, afin d'obtenir autant de fichiers .txt.mlc que de phrases, on utilise **mlc_en_mlcs_phrases.pl** qui génère autant de fichiers .txt.mlc qu'il y a de mlc dans le fichier issu directement d'EULER. Ce sont ces fichiers indexés qui se trouvent dans le dossier mlc_cocteau.

2.1.2.1) sdif_cocteau:

Ce dossier contient les fichiers .sdif regroupant les informations des unités segmentales. Ils sont générés automatiquement grâce au script MATLAB: **make_sdif.m**
Il suffit de spécifier les path du dossier contenant les fichiers .txt.mlc (mlc_cocteau) et de celui comportant les fichiers .diph (diph_cocteau).

Pour chacun des fichiers d'une paire (.diph et .txt.mlc du même index), on exécute **mlcdiphwrite.pl** qui donne en sortie un fichier .txt dont chaque ligne comporte un phonème. Cela permet d'obtenir deux transcriptions phonétiques: Une du texte (à partir du fichier .txt.mlc) et une de l'audio (à partir du fichier .diph).

Puis on compare ces deux transcriptions avec la commande **diff** d'UNIX afin d'obtenir un fichier temporaire .diff avec les différences entre les deux.

Enfin, ces informations sont « synchronisées » par un script PERL: **mlcaligntosdif.pl** qui fournie en sortie le fichier .sdif réunissant ces différentes représentations. Voir le rapport de stage d'Orsten Karki pour plus d'infos. On a mis à jour ce script PERL de manière à ajouter les descripteurs relatifs à la prosodie fournie par la MLC. Ceci fait, on a, en sortie, un fichier .sdif contenant:

XSPH la frame qui correspond à un semi-phone

XDUR la durée du semi-phone en secondes (float)

XPOS colonnes 1 : position dans la syllabe, 2 : posision dans le mot, 3 : position dans la phrase. (matrice 1*3 de float)

XPHO le phonème correspondant en X-SAMPA (texte)

XBEN colonne 1 : flag indiquant s'il s'agit d'un semi-phone de début de diphone. Colonne 2 : ...fin de diphone (matrice 1*2 de float)

XSYL la syllabe de provenance en X-SAMPA (texte)

XWRD le mot de provenance en X-SAMPA (texte)

XLEX le mot lexical de provenance (texte)

XGRN la nature grammaticale du mot de provenance (texte)

XFST colonne 1 : flag indiquant s'il s'agit du premier phone de la syllabe, colonne 2 : ce phone appartient-il à la première syllabe du mot ? Colonne 3 : appartient-il au premier mot de la phrase ? (matrice 1*3 de float)

XLST colonne 1 : flag indiquant s'il s'agit du dernier phone de la syllabe, colonne 2 : ce phone appartient-il à la dernière syllabe du mot ? Colonne 3 : appartient-il au premier mot de la phrase ? (matrice 1*3 de float)

Les informations sur les accents donnés par le fichier MLC:

XACC L'accent de la syllabe de provenance {NA, AF, UNDIFINED}

XTON Le ton de l'accent de la syllabe de provenance {L-L-,ll,HH,H+H+,P1,P2}

XSEN L'index de la phrase d'où provient l'unité.

Ces fichiers .sdif sont placés dans le dossier généré par **make_sdif.m:** sdif_cocteau

2.1.2.3) sdif_txt_cocteau:

C'est la copie conforme de sdif_cocteau mais en fichiers textes pseudo-sdif. (sdif en ASCII et pas en binaire). D'ailleurs il est crée en même temps que sdif_cocteau par **make_sdif.m.**
Il permet simplement de visualiser les fichiers .sdif sans utiliser **sdittotext.**

Les fichiers .sdif accompagnés des fichiers audios .aiff constituent déjà une base virtuelle. Il s'agit maintenant de l'importer dans la base « physique ».

Remarque concernant le script **mlcaligntosdif.pl** : CATERPILLAR donne la possibilité d'importer des descripteurs de toute provenance. Il aurait été plus judicieux alors d'importer les descripteurs une fois les unités créées. Cela aurait permis par exemple de changer un fichier .diph sans avoir à réimporter le fichier .aiff et les autres... Ce choix n'a pas été fait par Orsten qui préfère aligner les descripteurs hors de la base. Nous n'y reviendrons donc pas mais nous signalons simplement qu'une alternative est possible et est plus en adéquation avec le système pensé par Diemo Schwarz.

2.2) Insertion des unités dans la base:

2.2.1) La base TALKAPILLAR31:

Avant toute chose, il faut demander à Diemo Schwarz qu'il vous déclare en tant qu'utilisateur de la base.

On peut y accéder via PSQL. Pour cela il suffit de se logger sur KETHUK (seul ordi supportant PSQL)

> ssh kethuk

> password: XXXXXX

> psql talkapillar31

là on se trouve dans la base et on peut la visiter via des commandes PSQL du type:

> select * from basefile; (* = all)

> select * from unitfeature;

> select * from [toutes les tables de la base]

On va voir plus loin comment accéder aux noms de toutes ces tables via Matlab.

Un autre commande importante consiste à supprimer des infos dans la base: par exemple:

> delete from basefile where bfid = 46531;

Cette commande supprime le fichier correspondant à l'ID 46531.

De manière générale, il est bon de vérifier via PSQL les importations et autres manipulations que l'on effectue avec MATLAB via le DBI (DataBase Interface).

2.2.2) Création des unités « réelles » dans la base de données:

A partir de maintenant, la plupart des opérations s'effectuent dans MATLAB. Avant de le lancer, placez vous dans le répertoire /thesis/talkapillar/ (copiable via CVS), de manière à ce que MATLAB lance le fichier **startup.m** initialisant la connection entre MATLAB et PostGreSQL.

Préalablement, On a ajouté quelques descripteurs relatifs à la prosodie dans **insertalkapillardescriptors.m** et dans **insertalkapillarcategories.m** qui sont en quelque sorte des scripts d'initialisation de la base de manière à la formater pour la parole. Ainsi, il y existe des descripteurs et des catégories relatives aux représentations symboliques nécessitées plus tard lors de la sélection.

Puis l'importation se fait via le script matlab: **importall.m** qui se résume aux étapes suivantes:

Tout d'abord, il faut spécifier les différents path des dossiers comportant les fichiers audios (aiff_cocteau), les fichiers mlc (mlc_cocteau) et les fichiers fo.sdif (fo_sdif). Il faut aussi déclarer à quel corpus vont être attacher les unités. Si besoin est, il faut en créer un avec « >> dbi addcorpus ».

Puis MATLAB va chercher dans les dossiers des triplets de fichiers (qui doivent donc impérativement comporter dans leurs noms le même nombre et un seul). Il ajoute d'abord le Basefile qui est le fichier audio .aiff.

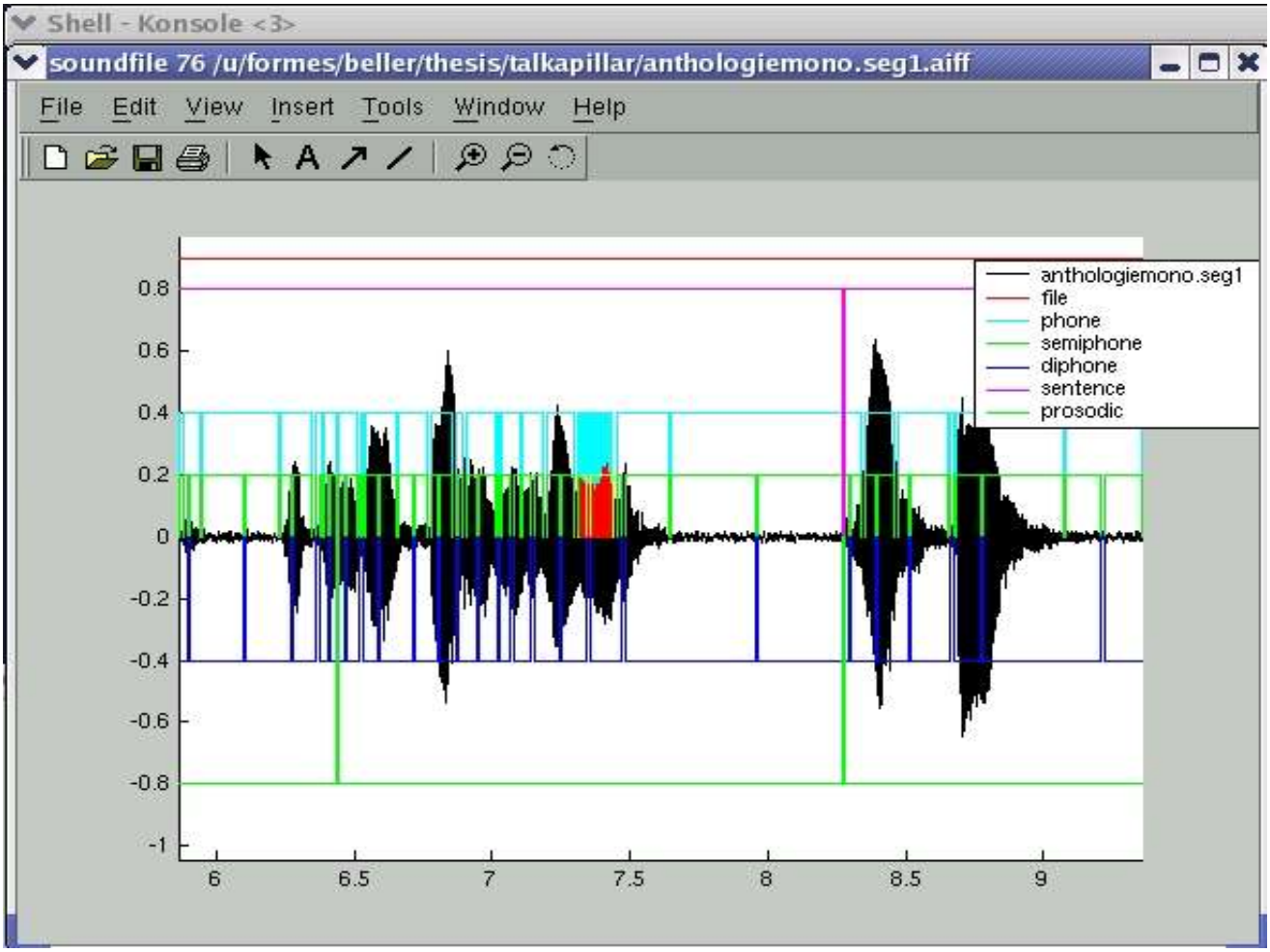
Après, il appelle la fonction **importphonemes.m** qui va lire le fichier .sdif et en extraire des unités correspondantes aux semi-phones, aux phones et aux diphones. Ce script MATLAB permet donc de fournir les unités segmentales à la base de données via le dbi. Chaque semi-phone est lié par un lien de parenté au phone et au diphone qui le contient.

Enfin, il crée des unités suprasegmentales à l'aide de deux scripts MATLAB:

- **addsentenceunits.m** : Crée des unités phrases (les unités les plus longues).
- **addprosogroup.m** : Crée des unités prosodiques selon la méthode citée précédemment.

  Attention: les unités doivent être importées dans l'ordre chronologique afin que les groupes accentuels soient bien construits. Ces unités supra-segmentales sont les parents des unités segmentales (semi-phones, phones et diphones).

On a alors une structure arborescente du type:



La hauteur des unités ici représentées équivaut à des degrés de parenté. Les couples d'unités successives et de même nature ont un lien previous/next. On voit très bien le décalage d'un semi-phone entre les phones et les diphones.

La figure ci-dessus a été obtenue grâce à « >> dbx corpus ». Cette commande permet de visualiser les unités et les descripteurs associés (seul fo est intéressant car non-statique). Elle permet aussi de les écouter afin de savoir si il n'y a pas décalage entre le phone et sa description symbolique. Malheureusement, il semble que la synchronisation des informations soit un point de fragilité du système. Afin d'y remédier, il faut améliorer **mlcaligntosdif.pl**.

Jusqu'à aujourd'hui, cette étape de vérification est manuelle et nécessaire.

Une autre amélioration de l'importation envisageable est dans son optimisation. Pour l'instant, l'importation d'une quinzaine de phrases prend 24H. Cela est trop long vis à vis du nombre de phrases que l'on souhaite importer. On en a aujourd'hui environ 1000, ce qui prend 2 mois environ !!!

2.3) Création d'une cible:

Afin de sélectionner des unités, il nous faut une phrase cible décrite de la même façon que ces unités: grâce aux mêmes descripteurs symboliques.

Bien entendu, il manque à la cible, une représentation acoustique et ce qui en découle (.fo.sdif et .diph). C'est cette représentation là que nous cherchons à créer grâce à la synthèse.

Nous n'en possédons qu'une description symbolique (.txt et .txt.mlc). L'importation d'une phrase cible s'effectue alors de la même manière que l'importation des unités sources. Après un passage par EULER pour avoir la mlc correspondante, on utilise successivement:

- **make_sdif_target.m:** Ce script crée des dossiers sdif_target et sdif_txt_target et appelle la script suivant
- **mlctosdif_target.pl:** Il crée un fichier .sdif à partir d'un fichier .txt.mlc. Il introduit dans les matrices dédiées aux marqueurs temporels de fausses valeurs. Ainsi chaque semi-phone possède la même durée: 0.1s. Ceci afin de pouvoir visualiser une cible de la même manière que l'on visualise une véritable segmentation.
- **Importtarget.m:** il effectue les mêmes taches que importall.m et en ajoute une: Il crée un fichier audio .wav qui ne comporte que très peu de zéro (fréq. d'échantillonnage = 10 Hz sur 8 bits). Ce faux fichier audio très léger, nous permet la visualisation avec « >> dbx corpus » des unités cibles. Il a une durée de la longueur de la phrase (0.1*nb de semi-phones). Il permet aussi une meilleure gestion des unités cibles dans la base. Il est à remarquer que TALKAPILLAR possède la possibilité de manier des fichiers virtuels. Nous avons préféré cette solution pour des raisons pratiques mais avons conscience de la supercherie. Enfin la taille de ces fichiers reste négligeable devant celle des nombreux fichiers sources.

Une fois cette cible créée, on maintenant capable de passer à l'étape suivante: La sélection d'unités.

3) Sélection des unités:

Pour connaître les algorithmes et leurs implémentations, je renvoie le lecteur à la thèse de Diemo Schwarz. Ce rapport étant principalement un guide technique de la démarche à suivre, nous n'y expliquons pas l'algorithme de Viterbi.

Afin de sélectionner des unités de la source en fonction de la cible, on se sert d'un seul script MATLAB appelé: **selectall.m**. Voici comment il procède:

Premièrement, on lui spécifie le type d'unités que l'on souhaite utiliser (semi-phone, phone ou diphone). Initialement, TALKAPILLAR est un système de sélection d'unités de tailles variables. Mais après avoir tester la concaténation de semiphones, nous nous sommes aperçus que le résultat n'était pas convaincant. On a donc borné le type d'unités dès le départ. Cela permet aussi de comparer les synthèses suivant que l'on concatène des semi-phones, des phones ou des diphones. Pour l'instant, il semble que le diphone soit le meilleur type d'unités pour la synthèse. De toute façon, il est possible de choisir le type semiphone et de régler les poids dans le coût de concaténation de manière à sélectionner des unités plus larges. Mais faute d'unités, il ne nous a pas été possible jusqu'alors de bien régler les poids dans la sélection. C'est d'ailleurs une étape très importante qu'il va falloir résoudre.

Puis on choisit les corpus dont on va se servir:

- corpus_pho_source: Le corpus dans lequel on va sélectionner les unités segmentales sources.

- corpus_pho_target: Le corpus dans lequel réside les unités cibles que l'on veut synthétiser.

- corpus_proso_source: Le corpus dans lequel on va sélectionner les unités supra-segmentales sources.

- corpus_proso_target: Qui doit être le même corpus que corpus_pho_target.

Après on écrit le nom du fichier cible que l'on souhaite synthétiser. MATLAB va alors chercher dans la base les groupes prosodiques cibles correspondant au fichier cible et créés lors de l'importation. Puis il appelle la fonction **selectprosogroup.m** afin de sélectionner dans corpus_proso_source, des unités supra-segmentales desquelles on va utiliser la courbe réelle de Fo. Cette étape de sélection se déroule de la même manière que celle que l'on va décrire ci-après. Nous ne rentrons donc pas dans le détail.

Une fois les unités prosodiques choisies dans le corpus source (valeurs de retour de la fonction **selectprosogroup.m**). On va chercher quels sont leurs « enfants » du type d'unité choisi au départ. Par exemple, si on a choisi de synthétiser par diphone, MATLAB va chercher les diphones source appartenant au groupe prosodique source précédemment sélectionné. Ceci n'est pas du tout une sélection d'unités segmentales, c'est simplement une étape nous permettant d'accéder au Fo moyen de chaque unités segmentales composant la courbe intonative que l'on souhaite reproduire.

En effet, nous préférons utiliser la valeur moyenne de Fo durant une unité pour construire une représentation de la courbe intonative moyenne de la phrase. Comme cela, on s'affranchit des

variations micro-prosodiques (variations intrinsèques à la prononciation d'un phonème) qui ne sont pas représentatives de l'expression véhiculée, mais propre au phénomène d'articulation.

Il est probable que les unités prosodiques choisies ne comportent pas le même nombre d'unités segmentales que les unités que l'on souhaitent synthétiser. Dans ce cas, on rééchantillone la courbe des valeurs moyennes de Fo d'un ratio: (Nb d'unités du prosogroup cible)/(Nb d'unités du prosogroup source). Ainsi, on obtient pour chacune des unités segmentales cibles une valeur de Fo réelle issue de la première étape de sélection. L'enchaînement de ces valeurs moyennes reflètent un contour réel que le locuteur à déjà prononcé.

Ainsi, on ajoute à la description symbolique des unités segmentales cibles, un descripteur acoustique (Fo) qui nous permet d'affiner la sélection d'unités segmentales. Celle-ci s'effectue en trois étapes:

- La première est la définition d'une structure appelée ici « phoneme ». Cette structure permet de définir quels descripteurs on va utiliser pour définir les fonctions de distances, de leurs attribuer un poids ainsi qu'une fonction d'évaluation(binaire, euclidienne...). On définit ces informations pour les deux fonctions de distances (à la cible et de concaténation) et leurs poids respectifs dans l'évaluation globale. On a la possibilité de présélectionner des unités en ajoutant une contrainte du type (Fo < 300 Hz), ce qui réduit l'espace de recherche et donc le temps de recherche par conséquent.

- La deuxième étape de la sélection et le chargement des unités des corpus choisis ainsi que leurs valeurs correspondant aux descripteurs définis préalablement. On charge ces sous ensembles des corpus (source et cible) dans des variables globales de manière a les réutiliser pour d'autres sélections si toutefois, rien a changer dans la définition de la structure de sélection « phoneme ».

- La troisième et dernière étape et la sélection en elle même. Réalisée par la fonction **unit_selection_pruned.m.** Par un algorithme de Viterbi, cette fonction sélectionne des unités en calculant des distances définit par la structure « phoneme », entre les unités des deux corpus choisis et chargés précédemment. Le vecteur donné en retour comportent les Id's des unités sélectionnées qu'il ne nous reste plus qu'à concaténer grâce à la fonction **concatenate.m.** C'est cette fonction qu'il va falloir changer de manière à utiliser DIPHONE pour la synthèse et ainsi pouvoir effectuer des transformations élémentaires sur les unités (Time Stretch et Pitch Schift).

4)Travail à faire:

- Améliorer et finaliser l'outil d'alignement et de segmentation
- Porter EULER sous UNIX
- Des erreurs de synchronisations induisant un décalage entre les unités acoustiques et leurs représentations symboliques doivent être rectifiées dans le script mlcaligntosdif.pl.
- Mettre en place les fonctions de coût, les algorithmes de sélection et les réglages des poids dans les choix des unités segmentales et suprasegmentales.
- Transformer en utilisant le croisement de données segmentales et suprasegmentales.
- Synthétiser en faisant le pont entre MATLAB et DIPHONE UNIX.
- Evaluer.
- Ajouter des descripteurs segmentaux.
- Ajouter l'énergie (intensité).
- Introduire un coup d'apparition: Ceci afin de faire varier les unités choisies. On les pénalise lorsqu'elles sont utilisées dans la phrase de manière à ne pas avoir trois fois le même « E » dans la même phrase.
- Optimiser l'importation des unités.
- Implanter l'outil d'alignement et de segmentation dans TALKAPILLAR, de manière à utiliser la sélection d'unités pour l'alignement ainsi que les unités déjà présentes dans la base. Ainsi, plus on en importera, plus on améliorera l'alignement.

Réglage des poids automatique ???

Conclusion:

Ce stage m'a permis de découvrir à quel point la prosodie est importante dans le signal de parole. Il est le point de départ et le support matériel d'une réflexion globale menée sur les liens entre la prosodie et la musique. Cette réflexion est détaillée dans un mémoire écrit durant cette année sous la direction d'Anne Sédés (Département musicologie de Paris VIII).

De plus, les diverses manipulations pour arriver à l'élaboration du système TALKAPILLAR, m'ont appris à me servir de divers outils: MATLAB, UNIX, PERL, PostGreSQL, DIPHONE...

L'étude sur la prosodie et la mise en place du système ont été pour moi, deux bonnes manières d'appréhender le métier de la recherche.

TALKAPILLAR, à mon sens, est le point de départ d'un bel outil qui lorsqu'il sera consolidé et validé permettra de nombreuses utilisations aussi bien pour le scientifique que pour l'artiste. En effet, de nombreuses perspectives sont permises, notamment dans le croisé et l'hybridation des données de parole et de signaux musicaux. Enfin, il est aussi un bel outil pour la classification et l'indexation et peut servir de base à une étude plus poussée concernant la prosodie et la parole en général.

Pour en savoir plus, je vous invite à lire le mémoire de musicologie intitulé: La musicalité de la voix parlée.

Bibliographie:

Système FESTIVAL: (pour la synthèse de l'anglais)

Dusterhoff, K. and Black, A. (1997). Generating F0 contours for speech synthesis using the Tilt intonation theory. Proceedings of ESCA Workshop of Intonation, September, Athens, Greece. Recommended: Most recent intonation module in Festival

Black, A. (1997). Predicting the intonation of discourse segments from examples in dialogue speech, ATR Workshop on Computational modeling of prosody for spontaneous speech processing. ATR, Japan. Republished in "Computing Prosody," eds. Y. Sagisaka, N. Campbell and N. Higuchi, Springer Verlag.

Black, A. and Hunt, A. (1996). Generating FO contours from ToBI labels using linear regression Proceedings of ICSLP 96, vol 3, pp 1385-1388, Philadelphia.Recommended: Description of syllable target model in Festival

Black, A. and Campbell, N. (1995). Predicting the intonation of discourse segments from examples in dialogue speech, (Short version) ESCA workshop on spoken dialogue systems, Denmark.

Black, A. (1995). Comparison of algorithms for predicting accent placement in English speech synthesis, Spring meeting of the Acoustical Society of Japan.

Unit selection:

Hunt, A. and Black, A. (1996). Unit selection in a concatenative speech synthesis system using a large speech database, Proceedings of ICASSP 96, vol 1, pp 373-376, Atlanta, Georgia.Recommended: Best published description of Hunt and Black unit selection technique.

Thèse de Diemo Schwarz (2004): Data-driven concatenative sound synthesis. IRCAM

Thèse de Romain prudon (2003): Synthèse de la parole multilocuteur par sélection d'unités acoustiques. LIMSI

Prudon R., d'Alessandro C. (2001). A selection/concatenation TTS synthesis system : Databases developement,system design, comparative evaluation. presented at Speech Synthesis Workshop 4th , Pitlochry, Schotland.

Prudon R., d'Alessandro C., et Boula de Mareüil P (2002). Prosody synthesis by unit selection and transplantation on diphones, presented at IEEE 2002 Workshop on speech synthesis, Santa Monica, USA.

Bozkurt B., Dutoit T., Prudon R., d'Alessandro C., et Pagel V (2002). Improving quality of MBROLA synthesis for non-uniform units synthesis, presented at IEEE 2002 Workshop on speech synthesis, Santa Monica, USA.

Thèse de Christophe Blouin

Prosodie:

Anne Lacheret-Dujour et Frédéric Beaugendre."la prosodie du Français". CNRS Langage. CNRS edition.

Langues, prosodie, syntaxe. Jacqueline Vaissière

Www.cavi.univ-paris3.fr/ilpga/ed/dr/jvdr2/articlesJV/vaissiereatala1997.pdf

Et de nombreux autres (voir le mémoire sur la musicalité de la voix parlée).