

TALN 2001

Vers un logiciel multilingue et gratuit pour l'aide aux personnes handicapées de la parole : HOOK (une interface du projet W)

Xavier RICCO, Thierry DUTOIT

Multitel TCTS – Faculté Polytechnique de Mons
Parc Initialis
7000 Mons Belgique
{ricco,dutoit}@tcts.fpms.ac.be
URL : <http://tcts.fpms.ac.be/synthesis>

Résumé – Abstract

Le but du projet W, lancé en 1999 par le laboratoire TCTS de la Faculté Polytechnique de Mons est de permettre aux personnes handicapées de la parole de profiter des développements récents en matière de synthèse vocale. C'est dans le cadre de ce projet qu'a été réalisé HOOK, un logiciel de désabréviation capturant toute frappe au clavier sous MS-Windows, et proposant automatiquement et en temps réel, à travers une fenêtre "popup", le résultat de la désabréviation à opérer. Les règles utilisées pour la désabréviation sont décrites par une grammaire régulière multi-niveaux. Une implantation de ces règles pour les abréviations Braille est fournie pour le français et l'anglais. Couplé au logiciel EULER de synthèse vocale à partir d'un texte (ou à tout autre logiciel de synthèse, moyennant adaptation des sources, fournies gratuitement) HOOK permet aux personnes handicapées de la parole de communiquer mot à mot grâce au synthétiseur vocal MBROLA (inclus dans EULER). HOOK, MBROLA et EULER sont distribués gratuitement pour usage non-commercial et non-militaire.

The goal of the W project, launched in 1999 by the TCTS Lab of the Faculté Polytechnique de Mons, is to let people with speech disability benefit from recent developments in terms of speech synthesis systems. Hook has been developed in the context of this project; it is a software that hooks keystrokes in MS-Windows, performs abbreviation expansion, and automatically shows, in real time and through a popup window, the proposed expanded form. Expansion rules are based on multi-level regular rewrite rules. An implementation of Grade II Braille abbreviations is provided for French and English. When used in combination with the EULER text-to-speech system (or with any other TTS system), HOOK makes it possible for people with a speech handicap to communicate through the MBROLA speech synthesizer (included in EULER). HOOK, MBROLA, and EULER are distributed for free, for non-commercial, non military use.

Mots clés

Abréviations, synthèse vocale, MBROLA, EULER, TTS, JSML.

Introduction

Le présent article a pour objectif de présenter un logiciel développé dans le cadre du projet W lancé en 1999 par le groupe de synthèse vocale de la Faculté Polytechnique de Mons (FPMs). Ce logiciel (HOOK), décrit ci-dessous, combine le synthétiseur vocal multilingue EULER (par le biais d'une interface en JSML) ainsi que les abréviations des mots les plus couramment utilisées dans une langue donnée.

En premier lieu, nous décrivons le projet W qui propose une interface adaptée entre la personne handicapée et la synthèse vocale. Nous présenterons ensuite le logiciel HOOK qui capture directement toute frappe au clavier, propose automatiquement un équivalent désabrévié puis effectue une synthèse vocale à travers l'interface JSML du logiciel EULER (Dutoit et al., 2000 ; Bagein et al., 2000).

1 Le projet W

Le projet W (Tounsi et al., 2000) cherche à faire profiter les personnes handicapées de la parole des développements récents en matière de synthèse vocale.

Dans cette optique, le logiciel W proposé permet aux personnes handicapées de bénéficier du synthétiseur vocal MBROLA (Dutoit, Pagel, 1996) pour participer en temps réel à des discussions orales. Le plus gros problème à résoudre pour atteindre cet objectif est celui de la vitesse de frappe : "comment utiliser un synthétiseur vocal pour discuter avec d'autres personnes sans les ennuyer sachant que le temps de frappe au clavier est nettement supérieur au temps de parole ?". Plusieurs solutions ont été testées pour palier ce problème. Elles s'articulent autour de la *prédiction et d'abréviation de mots*, et d'*écriture en sténographie*.

La prédiction de mots a déjà fait l'objet de nombreuses études, et plusieurs logiciels ont été développés dans ce cadre (voir la liste des adresses utiles sur la toile, en fin d'article). Ils ont à peu près tous la même fonctionnalité : en fonction des lettres déjà frappées et des mots contenus dans une base de données intégrée au logiciel, le programme prédit et affiche à l'écran les lettres manquantes, sans que l'utilisateur ait à les frapper. Une étude menée par le KTH, qui travaille depuis de nombreuses années sur la prédiction pour l'aide aux handicapés, a montré l'inefficacité relative de ce genre de méthode pour augmenter la vitesse de saisie du texte au clavier, et donc permettre aux personnes handicapées de communiquer en temps réel (Magnusson, 1994). La principale cause de ce manque d'efficacité provient du fait que le ou les mots prédits par la machine ne peuvent l'être en même temps par l'utilisateur humain. Ce dernier doit donc signifier son choix, ce qui requiert du temps et constitue une source d'erreurs, lesquelles sont également consommatrices de temps (de correction).

Si le but est d'optimiser la vitesse de saisie, il est donc intéressant d'utiliser un principe d'écriture abrégée : si la correspondance entre abréviations et mots du lexique est biunivoque, il est en principe possible à l'utilisateur de les mémoriser, ou du moins d'en mémoriser les règles (moyennant une certaine charge cognitive), de façon à éviter la nécessité une constante validation/invalidation des mots proposés, comme c'est le cas en prédiction.

Le projet W (Tounsi, 1998) utilise la technique d'abréviation de mots ou de groupes de lettres les plus couramment utilisés. Plutôt que de chercher à inventer son propre lexique abrégé, ou ses propres règles d'abréviation, il propose l'utilisation (détournée) de l'abrégé Braille grade II. Ce langage est utilisé depuis plusieurs décennies par les personnes aveugles pour lire et écrire rapidement des textes au clavier. Il a donc déjà fait ses preuves en ce qui concerne la réduction du temps de saisie d'un texte. Le Tableau 1 ci-dessous présente un exemple de différentes abréviations Braille II, qui donne une bonne idée de la simplicité des contractions. La charge cognitive est importante, mais il existe des méthodes d'apprentissage progressif de l'abrégé, comme par exemple celui d'E. Kommer (1993), dont nous nous sommes inspirés pour construire la base de données des abréviations en Français.

L'utilisation du Braille II n'est que suggérée dans W : deux fichiers de règles d'abréviations sont fournis (en français et en anglais), et le logiciel est ouvert à toute autre type d'abréviation. Il permet d'ailleurs à l'utilisateur de créer ses propres abréviations et de les ajouter à la base de données de règles gérée par le logiciel W. Ceci permet aux personnes handicapées d'adapter le logiciel à leurs besoins.

Abréviation	Mot
aff	affaire
bd	boulevard
c(certes
?q	enquête
gè	guère
idl	idéal
Idlm	idéalement

Tableau 1 - Exemples d'abréviation en Braille abrégé.

2 MLRR, règles de réécritures multi-niveaux

Le transducteur sous-jacent au projet W est MLRR (pour Multi-Layer Rewriting Rules). Il s'agit d'un transducteur basé sur des règles de réécriture multi-niveaux. MLRR a été conçu dans un cadre plus général que celui de l'abréviation. Il est utilisé dans EULER pour réaliser la transcription phonétique des phrases à lire.

MLRR transduit une entrée symbolique (une chaîne de caractères contenant des symboles appartenant à un alphabet d'entrée) en une sortie symbolique (dans un alphabet de sortie), en utilisant pour ce faire une grammaire régulière décrite en terme de règles de réécritures. Les "symboles" et "alphabets" doivent être compris ici comme des termes génériques. En effet, ils peuvent s'apparenter à des caractères, des phonèmes, des syllabes, des mots, des phrases, ... Les règles sont basées sur les règles de réécritures régulières classiques (développées à l'origine dans le contexte de la phonologie générative) :

$$A \text{ ---> } B / L _ R \quad (1)$$

Cette règle signifie que le symbole (ou la séquence de symboles) A dans la couche d'entrée est transduit(e) en le symbole (ou la séquence de symboles) B dans la couche de sortie quand il(elle) est entouré(e) par les symboles (ou les séquences de symboles) L et R dans la couche d'entrée.

MLRR apporte à ce formalisme de base une dimension multi-niveaux, en ajoutant simplement des conditions à la règle (1), provenant d'informations contenues dans d'autres couches synchronisées avec la couche principale :

$$A1 / L1 _ R1 ; A2 / L2 _ R2 ; \dots ; An / Ln _ Rn \text{ ---> } B \quad (2)$$

Une telle règle signifie que le symbole (ou la séquence) A1 dans la couche d'entrée principale est transduit(e) en le symbole (ou la séquence) B dans la couche de sortie lorsqu'il(elle) est entouré(e) par L1 et R1 dans la couche d'entrée principale, et lorsque les symboles (ou séquences) A2, ... , An (qui sont synchronisés avec A1 mais appartiennent aux autres couches) sont respectivement entourés par les symboles (ou séquences) L2 et R2, ... , Ln et Rn dans leurs couches respectives.

La mise au point de ce transducteur a été motivée à l'origine par les problèmes rencontrés en synthèse vocale, où l'utilisation d'un simple transducteur régulier (pour la phonétisation, par exemple), quoique possible, complique l'écriture des règles, et rend très difficile l'analyse du processus de transduction (typiquement : pour en détecter les erreurs et corriger les règles). Il est par ailleurs bien connu que les langues naturelles se prêtent difficilement à une formalisation purement régulière, ce qui a souvent motivé les chercheurs à augmenter leurs transducteurs d'actions et de variables d'état (comme par exemple dans les ATNs, augmented transition networks). RULESYS (Carlson, Granström, 1976), développé il y a vingt-cinq ans mais toujours en usage (y compris commercial) est un excellent exemple d'utilisation de transducteurs réguliers pour la synthèse vocale. La complexité des règles les rend affaire de spécialistes (du logiciel, et pas du problème traité). Elle est essentiellement due au fait qu'un nombre important d'informations de natures fort différentes (typiquement, morphologiques, syntaxiques, graphémiques, etc.) se retrouvent juxtaposées dans la chaîne traitée. A l'opposé, on a vu apparaître depuis une dizaine d'années des analyseurs où ces diverses informations se trouvent codées dans un treillis, sur des niveaux distincts (voir par exemple Van Leeuwen, , te Lindert, 1993). Les niveaux autres que le niveau d'entrée et le niveau de sortie servent en quelque sorte de zones de stockage d'informations complémentaires nécessaires à la transduction, mais de nature différente à celle des niveaux d'entrée et sortie. MLRR est une implémentation de ce principe.

Le tableau 2 donne le contenu du début du fichier de règles pour l'utilisation du Braille abrégé en français (un fichier similaire est disponible pour l'anglais). On y trouve, sous l'en-tête « [STRUCTURE] », le nom du transducteur (« Wabrev ») et des couches utilisées : « Abréviation » et « Résultat ». Vient ensuite, sous l'en-tête « [CLASS] » la définition d'un ensemble de classes de symboles, dont l'utilisation simplifie l'écriture des règles. Les règles elles-mêmes sont données sous le champ correspondant au transducteur qu'elles décrivent.

*Vers un logiciel multilingue et gratuit pour l'aide aux personnes handicapées de la parole :
HOOK (une interface du projet W)*

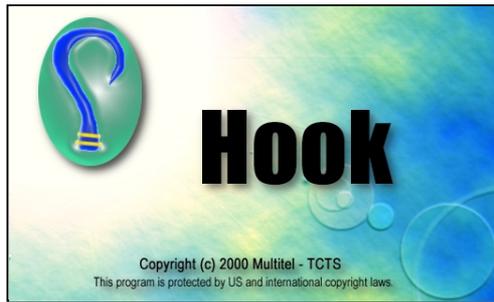
```
[INFO]
Règles pour la désabréviation des mots du logiciel Wabrev
[STRUCTURE]
WABREV = ABREVIATION -> RESULTAT
[CLASS]
CLASS <A> (a|e|i|o|u|à|â|é|è|ê|ë|î|ï|ô|ù|û|ü) # voyelles
CLASS <B> (b|c|ç|d|f|g|h|j|k|l|m|n|p|q|r|s|t|v|w|x|y|z) # consonnes
CLASS <P> (,|.|:|;|:) # ponctuations
CLASS <E> (√|!|\"|'|@|_|^|<|>|+) # caracteres speciaux faisant office de consonnes
CLASS <G> (√|!|\"|'|@|_|^|<|>|+|[]|/|\\|+ ) # caracteres speciaux remplaçant des voyelles
CLASS <V> (<A>|<H>|<G>)
CLASS <C> (<B>|<D>|<E>)
CLASS <L> (<V>|<C>) # toutes les lettres
CLASS <W> ( _ & | + ) # espace
CLASS <S> (^|<W>|<P>|[]|') # limite gauche d'un mot
CLASS <T> ($|<W>|<P>) # limite droite d'un mot
CLASS <X> (<T>|s<T>) # ajouter s si pluriel
CLASS <F> (<T>|e<T>) # existence du eventuellement pluriel I
CLASS <I> (es<T>|û<T>) # existence du eventuellement pluriel II
CLASS <Y> (<T>|s<T>) # ajoute s au mot
CLASS <Z> (<T>|m<T>) # ajoute ment au mot
[WABREV]
[](b)[]->(b)
[<S>](bo)[<X>]->(bonté)
[(<S>|<->)](bh)[(<X>|<->)]->(bonheur)
[<S>](bj)[<X>]->(bonjour)
[<S>](bd)[<X>]->(boulevard)
[<S>](bt)[<T>]->(bientôt)
[<S>](b*)[<X>]->(besoin)
[<S>](bê)[<X>]->(bête)
[<S>](bêm)[<T>]->(bêtement)
[<S>](bf)[<X>]->(bienfait)
[<S>](bfe)[<X>]->(bienfaisance)
[<S>](bf9)[<X>]->(bienfaiteur)
[<S>](bg)[<X>]->(budget)
[<S>](bgr)[<X>]->(budgétaire)
[](blt)[<T>]->(bilité)
...
[<C>](m)[<T>]->(ment)
...
[(<S>|ce|dé|er)](pd)[<F>]->(pendant)
...
[(<S>|re|dé|abs|con|tre|ob|sou|main|appar)](tn)[<T>]->(tenir)
...
```

Tableau 2 – Extraits d'une base de règles d'abréviation

On constate que les règles sont essentiellement de deux types : certaines réalisent la transduction d'un mot complet, les autres (comme « [](blt)[<T>]->(bilité) », qui spécifie que « blt » en fin de mot est transcrit « bilité ») sont de véritables règles contextuelles.

On constate également que l'utilisation faite de MLRR dans le projet W ne met pas encore à profit sa particularité « multi-niveau ». Elle s'avérera cependant intéressante lorsque W sera étendu dans le cadre du projet FASTY (voir section 4) pour intégrer, dans le processus de désabréviation, des informations morphologiques ou syntaxiques (les plus probables) sur les mots précédant le mot courant. Il sera alors possible, par exemple, de transcrire « blt » final en « bilités » lorsque le mot courant sera détecté (par des prédicteurs associés) comme probablement au pluriel.

3 Le logiciel HOOK



Le logiciel HOOK est une extension du logiciel W, qui constituait une interface rudimentaire entre l'utilisateur et MLRR. HOOK capture directement toute frappe au clavier sous MS-Windows, lance le transducteur MLRR (à chaque nouvelle frappe) et suggère la désabréviation obtenue à travers une fenêtre popup¹. Cette désabréviation peut être sélectionnée ou ignorée (voir figure 1).

Cette extension de W a l'avantage d'être active avec n'importe quelle fenêtre de texte sous MS Windows. Il a ainsi été possible de ne pas inclure directement le synthétiseur EULER dans HOOK. Au contraire, c'est l'utilisation d'EULER (en mode interactif) à travers HOOK qui rend la synthèse accessible aux personnes handicapées. HOOK gagne ainsi en versatilité (il est également utile pour des personnes qui ne souffrent pas d'un handicap vocal) et en compacité.

Bien que HOOK ne soit pas lié à un système de synthèse vocale particulier, l'utilisation d'EULER présente l'avantage d'une totale gratuité pour les personnes handicapées, et de la possibilité d'utiliser le synthétiseur sous Windows et Linux (et bientôt Macintosh). Moyennant une adaptation de HOOK à Linux, on pourrait ainsi toucher plus d'utilisateurs potentiels.

Dans la version actuelle, la synthèse est réalisée mot par mot : dès que MLRR a produit un mot complet accepté par l'utilisateur (la validation étant simplement matérialisée par l'enfoncement de la barre d'espace), ce dernier est envoyé au moteur de synthèse, qui le prononce de façon très (souvent trop) neutre. Il s'agit là d'une limitation dans le naturel de la voix produite, mais dont la portée est assez réduite. Les personnes chez qui le logiciel est en test souffrent en effet de handicaps tels que la simple prononciation, dans un temps raisonnable, des mots qu'elle veulent exprimer, constitue une amélioration considérable de leurs conditions normales de communication (le plus souvent, par articulation de lettres avec les mains, ou par écriture manuscrite, souvent difficile). Le logiciel offre par ailleurs la possibilité de répéter la phrase une fois écrite, en la dotant cette fois d'une intonation plus naturelle. Enfin, l'existence d'une trace électronique des messages produits est fort appréciée.

¹ Fenêtre qui apparaît automatiquement lorsqu'une touche est frappée.

Vers un logiciel multilingue et gratuit pour l'aide aux personnes handicapées de la parole : HOOK (une interface du projet W)

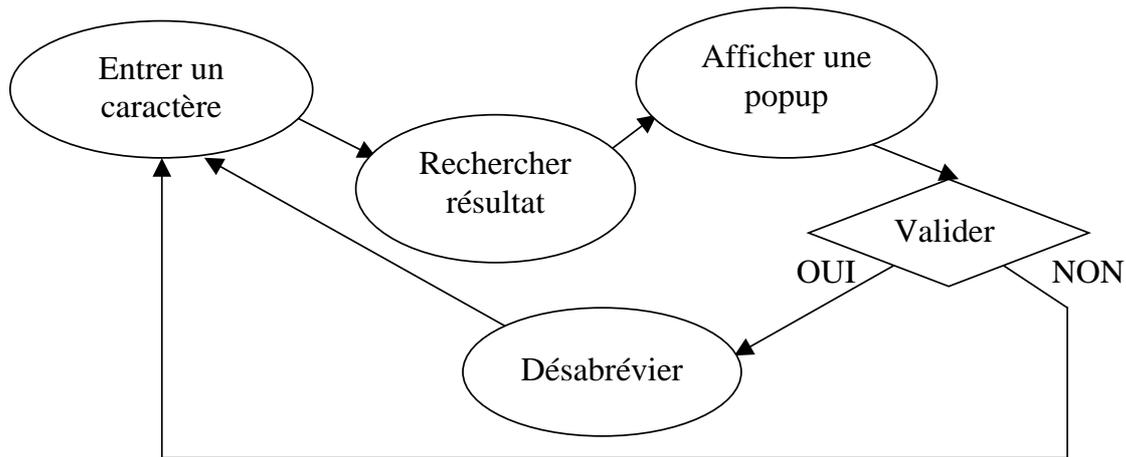


Figure 1 – Principe de l'interface du logiciel HOOK.

L'utilisation de HOOK est volontairement intuitive. Après avoir lancé l'exécution du logiciel, l'icône HOOK vient se mettre dans la « Traybar » de Windows, signifiant que HOOK est présent et actif. Cette icône est associée à un menu contextuel qui permet à l'utilisateur de changer de base de règles à tout moment, et autorise l'ajout de nouvelles règles (le plus souvent, ce seront des règles de remplacement d'une abréviation par un mot complet).

Il suffit de lancer une application gérant le clavier (comme les logiciels du courrier électronique, notepad, etc.). Dans le cas de notepad, par exemple (figure 2a, b, et c), l'entrée du caractère « b » fait apparaître le mot « bien » dans une fenêtre popup. Si l'utilisateur poursuit par l'entrée d'un « j », le logiciel propose « bonjour », et ainsi de suite jusqu'à la saisie d'un caractère d'espace, qui confirme la dernière abréviation proposée.

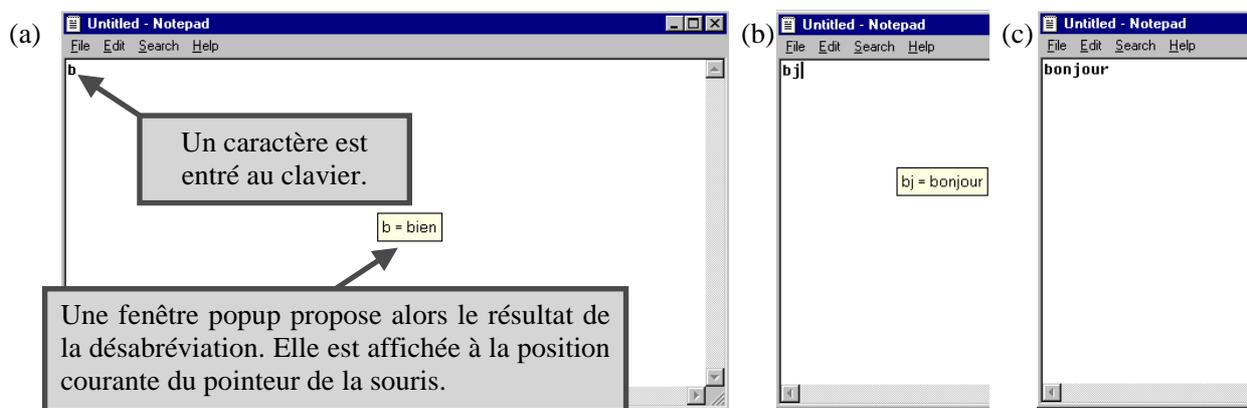


Figure 2 – Exemple de résultats obtenus avec HOOK.

4 Le projet FASTY

Le lancement du projet W nous a permis de participer au projet européen "Faster Typing for Disabled Persons" (n° IST-2000-25420), qui a débuté en janvier 2001.

L'objectif de FASTY est de créer un système permettant d'augmenter le taux de texte généré par seconde par les personnes handicapées. Ce système sera développé en allemand, en français, en néerlandais et en suédois. La conception, devra être transposable aux autres langues

européennes. FASTY apportera une assistance aux personnes handicapées ayant besoin de produire du texte plus rapidement, et ce avec moins de charge physique/cognitive et plus de grammaire et de vocabulaire. FASTY sera configurable pour différents types de handicaps, et différents paramètres de communication. Il permettra un meilleur accès aux traitements de textes, aux moyens de communication modernes et une utilisation plus rapide des synthétiseurs vocaux pour la communication. FASTY sera un système intelligent utilisant l'abréviation et la prédiction, favorisant ainsi un gain en terme de taux de frappe et plus spécialement pour les langues européennes qui sont hautement infléchies.

Nous intervenons dans ce projet en tant que fournisseur de technologie pour l'abréviation et la synthèse vocale, ainsi qu'en tant qu'apporteur pour le français et le néerlandais.

5 Participations au projet

La participation externe au projet W peut être de deux types : en temps qu'utilisateur ou développeur.

Le logiciel présenté ci-dessus (et qui sera placé en démonstration lors de l'ATELIER "handicap" de TALN 2001) est disponible gratuitement pour un usage non militaire et non commercial (suivant les termes d'un accord de licence accompagnant le programme). Nous demandons à ses utilisateurs de nous envoyer leurs commentaires sur sa facilité d'utilisation, afin d'améliorer l'interface et d'évaluer la méthode d'abréviation choisie.

Toutes les sources sont également téléchargeables depuis notre site Internet, ceci pour permettre à des développeurs extérieurs de mettre au point d'autres langues (tant au niveau de l'abréviation qu'à celui de la synthèse vocale), de faire le portage des différents modules sur d'autres plate-formes.

Remerciements

Nous tenons tout particulièrement à remercier Nawfal Tounsi, l'auteur initial de W, pour son soutien lors de la réalisation de HOOK. Merci également à Alain Ruelle, sans qui HOOK n'aurait pas vu le jour. Merci enfin à Olivier Platteau, pour son optimisme communicatif.

Références

Bagein M., Dutoit T., Malfrère F., Pagel V., Ruelle A., Tounsi N., Wynsberghe D. (2000), "Le Projet EULER : la synthèse de la parole générique multilingue", Actes des XXIIIèmes Journées d'Etudes sur la Parole, Aussois, pp.101-104.

Dutoit T., Bagein M., Malfrère F., Pagel V., Ruelle A., Tounsi N., Wynsberghe D. (2000), "EULER : an Open, Generic, Multi-lingual and Multi-Platform Text-To-Speech System", Proc. LREC'00, Athens, pp.563-566.

Dutoit T., Pagel V. (1996), "Le projet MBROLA : vers un ensemble de synthétiseurs vocaux disponibles gratuitement pour utilisation non-commerciale.", Actes des Journées d'Etudes sur la parole, Avignon, pp.441-444.

Projet EULER, " <http://tcts.fpms.ac.be/synthesis/euler> "

Magnusson T. (1994), "Evaluation of Predict", *Quarterly Progress Scientific Report*, KTH Department of Speech, Music and Hearing.

Projet MBROLA, " <http://tcts.fpms.ac.be/synthesis/mbrola> ". Carlson, R., Granström, B. (1976), "A Text-to-Speech System Based Entirely on Rules", *Proceedings of ICASSP 76*, Philadelphia, pp. 686-688.

Projet FASTY, " <http://www.fortec.tuwien.ac.at/reha.e/projects/fasty/fasty.html> "

Kommer E. (1993), "La méthode d'abrégé braille en noir", disponible auprès de l'association Valentin Haüy à Paris.

Tounsi N. (1998), "Mise au point d'un système d'aide aux handicapés de la parole utilisant un interpréteur de braille abrégé", Travail de fin d'études, Faculté Polytechnique de Mons.

Tounsi N., Dutoit T., Pagel V., Bagein M., Wynsberghe D., Ruelle A., Malfrère F. (2000), "Vers un logiciel multilingue et gratuit pour l'aide aux personnes handicapées de la parole : le projet W", Actes de *Handicap 2000*, Paris, pp.109-111.

Projet W, " <http://tcts.fpms.ac.be/synthesis/w> ".

Van Leeuwen, H.C., te Lindert, E. (1993), "Speech Maker: a Flexible and General Framework for Text-to-Speech Synthesis, and its Application to Dutch", *Computer, Speech and Language*, n°2, pp. 149-167.

Adresses Utiles sur la Toile

Accès à l'ordinateur pour les personnes ayant une incapacité physique :
" http://www.irdpq.qc.ca/technologies/tech_ordinateur.html "

Désabréviation :

InstantText, " <http://www.twsolutions.com/> "

Shorthand, " <http://www.shorthand2000.com/shorthand.htm> "

AAC Intervention, " <http://www.aacintervention.com/write.htm> "

Liste d'autres logiciels commerciaux " <http://www.disserv.stu.umn.edu/TC/Grants/COL/product/ComputerAccess/Keyboards/Software-AbbreviationExpa.html> ",

" <http://www.cs.wright.edu/bie/rehabengr/Comp1/Lang.htm> "

Prédiction :

KeyRep Word Prediction, " <http://www.nanopac.com/Keyrep.htm> "

Telepathic 2000, " <http://www.next-wave-solutions.com/tel20predtyp.html> "

Penfriend, " <http://www.spectronicsinoz.com/spectronics/product.asp?product=134> "

ProtoType, " <http://www.zygo-usa.com/prototype.html> "

TextHelp, " <http://www.dyslexic.com/help.htm> "

Winspeak, " <http://www.sensorysoftware.com/winspeak.html> "

WiVik, " <http://www.wivik.com/html/downloads.htm> "

Liste d'autres logiciels, " <http://www.utoronto.ca/atrc/reference/tech/wordpred.html> "