# Architecture et fonctionnement du système DIRA. De l'acoustique aux niveaux linguistiques

## *DIRA system architecture : from acoustics to linguistics*


J. CAELEN

ICP/INPG - UA CNRS n° 368
46, av. F. Viallet
38031 GRENOBLE CEDEX

J. Caelen est ingénieur de l'ENSHEIT de Toulouse et a obtenu successivement le doctorat-ingénieur en 1974, le doctorat d'état en 1979 spécialité Informatique et l'habilitation à diriger des recherches en 1986 à l'université de Toulouse. Il est actuellement responsable de l'équipe « Décodage et Compréhension de la parole » à l'ICP (Institut de la Communication Parlée) de Grenoble. Ses centres d'intérêt touchent tous les domaines de l'analyse à la compréhension de la parole.

M. K. NASRI

ICP/INPG - UA CNRS n° 368
46, av. F. Viallet
38031 GRENOBLE CEDEX

M. K. Nasri est ingénieur de l'école « Electricité et Mécanique » de Damas (Syrie). Il a passé 3 ans à l'ICP où il a obtenu le doctorat-INPG en 1990 dans la spécialité « Signal, Image, Parole ». Il est actuellement chercheur au CERS (Centre d'Etude et de Recherche Scientifique) à Damas. Il s'intéresse à la reconnaissance et à la compréhension de la parole, à l'I.A., au raisonnement cognitif, aux niveaux linguistiques et à la prosodie.

E. REYNIER

ICP/INPG - UA CNRS n° 368
46, av. F. Viallet
38031 GRENOBLE CEDEX

E. Reynier a soutenu une thèse de l'INPG de Grenoble en 1990 dans la spécialité « Signal, Image, Parole ». Il poursuit actuellement ses recherches post-doctorales à l'ICP dans le cadre du projet ESPRIT II « Multiworks ». Ses centres d'intérêt sont : reconnaissance et compréhension de la parole, I.A. et génie logiciel, linguistique computationnelle.

H. TATTEGRAIN

ICP/INPG - UA CNRS n° 368
46, av. F. Viallet
38031 GRENOBLE CEDEX

H. Tattegrain est ingénieur de l'ICPI de Lyon. Elle a obtenu le doctorat-INPG de Grenoble en 1990 dans la spécialité informatique et poursuit actuellement ses recherches post-doctorales à l'ICP où elle approfondit notamment le décodage acoustico-phonétique. Elle s'intéresse plus particulièrement à la reconnaissance de la parole, au traitement du signal, à la reconnaissance des formes, aux systèmes experts et à l'analyse des données.

# Architecture et fonctionnement du système DIRA


## RÉSUMÉ

Cet article décrit l'architecture et le fonctionnement du système de reconnaissance de la parole DIRA (DIRA : Dialogue Intégré et Reconnaissance Automatique) dans son état actuel. Ce système est un système multi-experts supervisé. Le superviseur organise les tâches de ses experts qui sont attachés aux diverses sources de connaissances : acoustico-phonétiques, lexicales, syntaxico-sémantiques, prosodiques et pragmatiques. Le tableau noir sert de boîte à lettre pour la communication de messages entre les divers modules ainsi que de mémoire à long terme où toutes les hypothèses en cours de construction sont consignées. Le superviseur est un planificateur opportuniste : il raisonne sur les données présentes dans le tableau noir et « calcule » la stratégie la meilleure pour activer les experts.

Les experts sont également décrits dans cet article : les DAP (décodages acoustico-phonétiques) avec leurs bases de connaissance représentées sous forme de règles qui contrôlent les transitions d'un ATN (Augmented Transition Network), les analyseurs linguistiques utilisant aussi le concept d'ATN compilé et la notion de grammaire lexicale fonctionnelle, la compréhension fondée essentiellement sur le phénomène d'amorçage sémantique et enfin l'analyseur prosodique à base de règles.

La mise en œuvre de ce système est commentée à travers des exemples et les résultats de reconnaissance sont discutés.



**MOTS CLÉS**

Reconnaissance et compréhension de la parole. Intelligence Artificielle et systèmes multi-experts. Décodage acoustico-phonétique. Stratégies de raisonnement. Linguistique computationnelle. Prosodie.



## *ABSTRACT*

*This article describes the architecture and the operation of the DIRA (Integrated Dialogue and Automatic Recognition) continuous speech recognition system in its present stage of development. The DIRA system is a supervised multi-expert system. The supervisor dynamically arranges the tasks of its expert modules, which are each attached to one of the subdomains of the speech recognition problem, i.e. the acoustic/phonetic, the lexical-, the syntactic/semantic-, the prosodic- and finally the pragmatic domain. A blackboard serves as message interchange medium between these expert modules, as well as long-term memory for the speech recognition process as a whole. The supervisor is an opportunistic planner : it reasons on the data present at the blackboard and « calculates » the best strategy (a scheme for the activation the expert modules) to resolve the current problem.*

*The operation of the individual expert modules is also addressed in this article : the APD's (Acoustic-Phonetic Decoders) with their knowledge bases represented as rules controlling the transitions in ATN's (Augmented Transition Networks), the linguistic analyzers using the same ATN concept and the principle of functional lexical grammars, the comprehensive analyzer founded on the principle of lexical priming and finally the rule-based prosodic analyzer.*

*The operation of the speech recognition system is commented, while providing examples and test results.*



***KEY WORDS***

*Speech Recognition and Understanding. Artificial Intelligence and Multi-Expert Systems. Acoustic-to-Phonetic Decoding. Reasoning and Problem Solving. Linguistics. Prosody.*


## Introduction

### OBJECTIFS

Cet article a pour but de décrire l'état actuel du système DIRA (DIRA = Dialogue Intégré et Reconnaissance Automatique de la parole) qui, à terme, sera un système de dialogue oral homme-machine complètement intégré. Pour le moment la composante pragmatique est encore absente du système ; celui-ci comporte cependant tous les modules permettant la compréhension de phrases à partir d'entrées acoustiques. Le domaine d'applications visé par DIRA concerne les langages de commande ou de communication à vocabulaire moyen (moins de 5 000 mots) et à syntaxe pseudo-naturelle.

### GÉNÉRALITÉS : ASPECTS COGNITIFS ET INTELLIGENCE ARTIFICIELLE

En psychologie, la nouvelle théorie mentaliste accorde une part importante au processus descendant, c'est-à-dire au contrôle par le mental sur les formes d'activité des composants neuronaux : dans cette théorie, les faits sont non seulement contrôlés du bas vers le haut (comme dans les modèles béhaviouristes) par l'activité microscopique (locale), mais aussi du haut vers le bas par action mentale descendante d'ordre macroscopique (globale). Ceci reviendrait, pour un modèle informatique tentant d'imiter le comportement cognitif humain, à posséder un module de raisonnement de haut niveau et un mécanisme de « conscience » c'est-à-dire de raisonnement sur lui-même. Une solution peut consister en un système modulaire — évidemment à stratégie mixte, ascendante et descendante — constitué de spécialistes [45], [31] et d'un module [23] (une sorte de superviseur ou de résolveur de problèmes) raisonnant à la fois sur les données fournies par ces spécialistes et sur sa propre stratégie (donc sur lui-même).

Dans un système de compréhension automatique de la parole, cette conception des choses a plusieurs conséquences :

(*a*) elle replace toutes les sources de connaissance au même niveau : le lexique ou la syntaxe ne jouent plus aucun rôle privilégié dans la stratégie de reconnaissance,

(*b*) un module de plus haut niveau, de type résolveur de problèmes, doit gérer la stratégie générale, et il doit le faire de manière dynamique, au regard de ce qui vient d'être dit ci-dessus, c'est-à-dire :

(*ba*) en activant ces spécialistes de manière opportune en fonction de l'évolution de la situation et des résultats partiels acquis et

(*bb*) en analysant son propre fonctionnement pour remettre éventuellement en question sa propre stratégie.

En reconnaissance et compréhension de la parole, la plupart des systèmes actuels orientés-connaissance [42], [18], [44], [9], [27], [46] sont constitués d'une communauté d'experts [30] (ou d'agents [37]) qui échangent leurs informations sous le mode vérification/proposition. Ces systèmes se distinguent entre eux surtout par la stratégie mise en œuvre plus que par la nature des sources de connaissances utilisées : en effet, ils manipulent presque tous des informations sur les niveaux acoustique, phonétique, lexical, prosodique, syntaxique et sémantique. Dans ces systèmes, les experts — associés aux sources de connaissance précédentes — fonctionnent de manière coordonnée (C) — sous la direction d'un superviseur — ou de manière autonome [13] (A) voire distribuée (D) [20]. Dans le premier cas (C), la stratégie est déterminée par un processus centralisé qui active les experts dans un ordre statique pré-établi ; dans le second cas (A + D), les compétences sont distribuées entre les experts (acteurs, agents) qui doivent coopérer entre eux de manière harmonieuse en se posant et résolvant leurs propres problèmes au moment opportun : ils n'ont à chaque instant qu'une connaissance partielle de leur environnement (ils ne connaissent pas les décisions prises par les autres agents au même moment). Des architectures plus récentes [26] utilisent des notions mixtes en introduisant deux catégories d'agents : ceux qui gèrent les connaissances et ceux qui les traitent.

Dans les systèmes « tableau noir » tels que Hearsay II [19], [38], [34], [29] les experts sont guidés par les données : en pratique, c'est l'organisation qui semble la plus attrayante puisque leur coordination devient théoriquement inutile. En réalité, du fait que les sources d'erreur se propagent entre tous les niveaux, un expert donné, devrait donc être capable de remettre en question ses propres résultats, d'affiner, voire modifier ses prédictions : or il n'a pas tout seul et au moment opportun, les éléments ni la compétence pour le faire (par exemple, certaines décisions de nature phonétique dépendent d'informations lexicales et prosodiques et réciproquement). Dans Hearsay II comme dans tous les systèmes « autonomes » les données doivent être alors corrigées — ou les hypothèses élargies et les mêmes experts réactivés plusieurs fois, ce qui peut provoquer un processus de bouclage et donc d'impasse puisque les sources de connaissance ne sont pas indépendantes. Une architecture d'agents complètement autonomes n'est pas souhaitable non plus pour le traitement de la parole si l'on veut introduire une notion de « conscience » telle que celle qui est suggérée dans les lignes précédentes : cette conscience est par nature un processus centralisé.

Dans les systèmes hiérarchisés tels que HWIN [57], [38], [34] ce risque de bouclage ne se produit pas mais la stratégie générale reste figée, ce qui conduit parfois à une inadéquation des traitements par rapport aux données présentes et à un manque de souplesse certaine dans la gestion des hypothèses et des tests.

Dans tous ces systèmes il manque donc, eu égard au comportement cognitif humain, un véritable niveau de raisonnement — dynamique et conscient — indépendant des sources de connaissances et des experts, pour gérer au mieux la stratégie générale de reconnaissance. C'est ce que nous avons tenté d'introduire dans le système DIRA, à travers une architecture supervisée par un planificateur opportuniste [54], c'est-à-dire :

(*a*) en utilisant la technique du tableau noir pour les nombreux avantages qu'elle présente comme le partage des données (mémoire commune à long terme) qui autorise le parallélisme des tâches et permet au superviseur de raisonner sur l'ensemble des données présentes à un moment donné ;

(*b*) en faisant jouer au superviseur un rôle « d'expert en stratégie » pour guider les experts du domaine parole. Cela revient à considérer que le superviseur est un planificateur qui organise le travail de ses experts qui trouvent toutes les ressources dont ils ont besoin dans le tableau noir. Chaque expert garde une autonomie dans son domaine, mais perd toute responsabilité dans le processus de décision général. En situation « normale », le système est guidé par les données tout en restant en permanence sous la vigilance du superviseur. Dans une situation « d'impasse » celui-ci impose une stratégie en fonction de la difficulté à résoudre.

Dans un véritable système adaptatif, le superviseur « calculerait » lui-même une nouvelle stratégie en fonction des événements survenus : pour le moment plusieurs stratégies lui sont données sous forme de règles par l'expert humain et des règles de choix (méta-stratégie) permettent de décider entre les stratégies possibles en fonction des événements extérieurs. L'aspect dynamique est ainsi sauvegardé dans cette première version du système.

# 1. Le système DIRA

## 1.1. ARCHITECTURE GÉNÉRALE

Le système DIRA est un système multi-experts organisé autour d'une architecture de tableau noir et d'un superviseur (*fig.* 1) — le terme multi-experts est utilisé ici dans le sens où pour chaque source de connaissance on peut considérer qu'il y a au moins deux expertises distinctes : l'une en prédiction ou proposition et l'autre en vérification ; ces deux expertises nécessitent des bases de connaissances et des mécanismes de résolution différents. Chaque expert a ses propres bases de connaissance et de faits (voir §§ 3, 4, 5, 6). Le superviseur peut être également considéré comme un expert dans la mesure où il a une base de faits et de règles (qui peuvent être obtenues par expertise humaine ou par raffinement de plans [18]) et un moteur d'inférences. Sa base de connaissance contient les plans procéduraux (règles) et les plans déclaratifs nécessaires à la planification, sa base de faits contient les problèmes posés et les variables d'environnement.

## 1.2. LES EXPERTS DU DOMAINE

Les experts du domaine de la parole sont :

— les **Décodeurs Acoustico-Phonétiques** (D.A.P.) qui proposent et vérifient des macro-traits et des traits phonétiques à partir du (ou sur le) signal d'entrée (voir § 3),

— les **Analyseurs Lexicaux** (A.L.) qui par des accès variés au lexique proposent et vérifient des mots (voir § 4),

— les **Analyseurs Syntactico-Sémantiques** (A.S.S.) ascendant et descendant qui contrôlent la cohérence des groupes syntagmatiques au niveau syntaxique et sémantique et prédisent le ou les prochains mots possibles (voir § 4),

— le **module de Compréhension** (C) qui contrôle les groupes de sens en établissant des relations sémantiques entre les constituants de la phrase (voir § 5),

— l'**Analyseur Prosodique** (A.P.) qui positionne des marques de débuts et de fins de mots sur le signal (voir § 6),

— le **Dialogue** qui assure la cohérence du niveau pragmatique à travers la communication utilisateur/application (il n'est pas encore intégré au système).

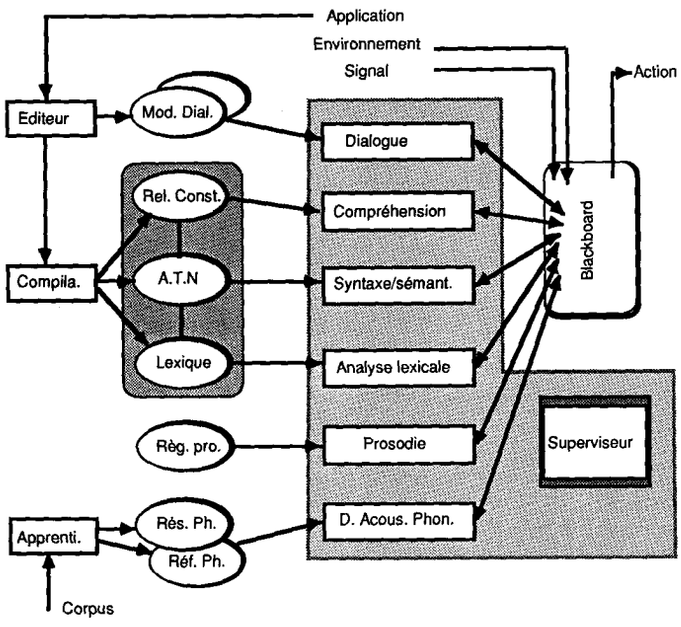



**Fig. 1. — Architecture du système DIRA. Mod. Dial. : modèle de dialogue (et autres bases non explicitées ici), Rel. Cons. : relations entre les constituants, ATN : réseau syntactico-sémantique à transitions augmentées, D. Acous. Phon. : décodeur acoustico-phonétique. Rés. Ph. : réseaux acoustico-phonétiques, Réf. Ph. : références phonétiques pour la quantification vectorielle.**

### 1.3. LE TABLEAU NOIR

Les experts communiquent avec le superviseur mais n'échangent pas directement d'informations entre eux : ils partagent seulement les données contenues dans le tableau noir. Ce tableau noir est constitué de listes hiérarchisées et de différents drapeaux qui signalent les étapes d'exécution du plan et l'état des experts — actif ou inactif.

Les listes du tableau noir sont :

**Liste d'hypothèses synt./sém. :** contient toutes les catégories syntaxico-sémantiques développées par les ASS.

**Liste d'hypothèses lexicales :** contient tous les mots-hypothèses développés par les AL.

**Liste de réussite :** contient les mots acceptés à la fin de l'étape de développement lexical et dont la séquence est compréhensible.

**Liste de liaison :** contient les phonèmes de liaison possibles entre 2 mots acceptés.

**Liste d'échec :** contient les mots abandonnés à la fin de l'étape de développement lexical ou qui ne sont pas compréhensibles.

**Liste de phonèmes :** contient les phonèmes proposés par les DAP ou par les AL.

**Liste de marqueurs prosodiques :** contient les étiquettes de début et fin de mots positionnées par l'analyseur prosodique.

Les drapeaux sont :

**Impasse** = « **vrai** » si *fin de fichier* est atteinte sans solution OU *listes d'hypothèses et de réussite* sont vides OU échec dans le *retour-arrière*.

**Parole** = « **vrai** » si *liste des phonèmes* est non vide.

**Phrase-reconnue** = « **vrai** » si *but final* est atteint.

**Séquence-compréhensible** = « **vrai** » si une *séquence de mots* subit avec succès le test de compréhension.

**Prédiction-liaison** = « **vrai** » si une *liaison* entre deux mots est possible.

**Liaison** = « **vrai** » si *liste de liaison* est non vide, c'est-à-dire si une liaison est attestée.

**Actif(X)** = « **vrai** » si l'expert X est actif.

### 1.4. LE MODÈLE DE COMMUNICATION

Les experts communiquent avec le superviseur par envoi de messages. Ces messages sont de deux sortes : (*a*) du superviseur vers l'expert X pour indiquer à ce dernier les tâches à exécuter, le mode et les contraintes d'action, (*b*) de l'expert concerné au superviseur pour lui communiquer les variables de contrôle et d'interruption.

## 2. Fonctionnement général

### 2.1. LE SUPERVISEUR

Il planifie les tâches des experts. La stratégie générale est de type gauche-droite avec retour arrière, gérée par points de rendez-vous ; ces points de rendez-vous sont des points de repli ou des points de reprise à partir desquels un nouvel essai est tenté dans une situation d'impasse. Toutes les solutions (opportunités) sont développées en largeur d'abord. Le superviseur guide pas à pas le déroulement de la reconnaissance en envoyant des messages d'activation à ses experts. Nous noterons :

X($m, f$) le message envoyé par le planificateur à l'expert X pour lui spécifier d'exécuter une tâche sur le mode $m$ et sur la fenêtre $f$ avec :

X nom générique de l'expert,

$m[p, v, pr, f, c\ ]$ $p$ = proposition, $v$ = vérification,

*pr* = retour au dernier point de rendez-vous, *f* = filtrage, *c* = collection d'informations,

*f* = [ − , = , + ] « − » = passé, « = » = présent, « + » = futur.

Les messages passés aux experts sont plus précisément :

**DAP(*p* =)** activation du DAP « proposition » pour proposer la liste des phonèmes les plus probables sur la fenêtre de signal actuelle. La tâche de l'expert est alors de lire la tranche de signal courante et de la décoder. Pour cela il fournit en sortie une liste de macro-traits phonétiques (éventuellement vide) et une liste de traits (idem), par exemple, V.ouv (« voyelles ouvertes »), Z.nil (« fricatives sonores »), etc.

**DAP(*v* =)** activation du DAP « vérification » pour tester sur le signal les phonèmes présents dans la liste hypothèse du tableau noir. Ces phonèmes ont pu être proposés par l'analyseur lexical dans une phase antérieure ou par DAP(*p*), DAP(*v*) ordonne la liste des phonèmes hypothèses selon un score décroissant. On peut remarquer que DAP(*v*) à qui on donnerait en entrée une liste ouverte à vérifier, fonctionnerait comme DAP(*p*) mais de manière descendante. En ce sens DAP(*p*) et DAP(*v*) sont concurrents.

Pour les autres fenêtres, passé et futur, le fonctionnement des experts est analogue.

**AL(*p* =)** activation de l'analyseur lexical « proposition » pour prédire tous les mots qui correspondent aux catégories syntaxiques affichées dans le tableau noir et qui contiennent les phonèmes hypothèses actuels. Les mots produits sont alors enregistrés dans la liste d'hypothèses lexicales.

**AL(*f* =)** activation du filtre lexical pour épurer la liste des mots hypothèses qui ne comportent pas suffisamment de phonèmes corrects,

**AL(*c* =)** activation de l'analyseur lexical en « proposition descendante » pour collecter les phonèmes pointant sur la fenêtre présente et contenus dans la liste des mots hypothèses. Les phonèmes ainsi extraits sont mis avec leur contexte dans une liste de phonèmes en attente d'une vérification par DAP(*v*) par exemple.

**AL(*psc* =)** activation de l'analyseur lexical en « proposition large » pour élargir le champ de recherche lexicale en cas d'impasse locale.

**AL(*v* =)** activation de l'analyseur lexical en « vérification » pour confirmer qu'une suite de phonèmes donnée peut faire partie d'un mot du vocabulaire.

Pour les autres fenêtres, passé et futur, le fonctionnement des experts est analogue.

**ASS(*p* =)** activation de l'analyseur syntaxico-sémantique en proposition (analyse ascendante) pour chercher les catégories lexicales possibles pouvant succéder à chaque mot reconnu. L'analyseur pose systématiquement un point de rendez-vous après chaque mot reconnu et dont la compréhension est assurée.

**ASS(*p* +)** activation de l'analyseur syntaxico-sémantique en proposition (analyse descendante) pour développer les hypothèses syntaxiques après une fin de mot non encore atteinte.

**ASS(*pr* −)** activation de l'analyseur syntaxico-sémantique pour retourner au point de rendez-vous précédent en modifiant les drapeaux dans le tableau noir et en filtrant les listes d'hypothèses.

**ASS(*v* =)** activation de l'analyseur syntaxico-sémantique pour vérifier qu'une suite de mots s'arrêtant à la fenêtre courante, est syntaxiquement et sémantiquement correcte.

**AP(*p* =)** activation de l'analyseur prosodique pour positionner sur le signal les débuts et fins de mots lexicaux et grammaticaux, les pauses, les débuts et fins de syntagme.

**AP(*f* =)** activation de l'analyseur prosodique pour filtrer les phrases candidates correctes à l'aide des marqueurs prosodiques.

**C(*v* =)** activation du module de compréhension pour vérifier si la liste courante mais encore partielle des mots hypothèses est compréhensible.

Dès qu'il reçoit un message, l'expert X exécute la tâche qui lui est demandée de la manière suivante :

- ☐ 1. Lecture du tableau noir : mise à jour de sa base de faits en fonction du contexte, sélection des problèmes à traiter et de leur degré de profondeur, lecture des hypothèses en cours et des contraintes locales.
- ☐ 2. Résolution du(es) problème(s) posé(s).
- ☐ 3. Écriture dans le tableau noir des résultats (hypothèses et scores), des drapeaux et des variables modifiées.
- ☐ 4. Désactivation (avec envoi de message au superviseur) et attente d'un autre problème à traiter.

### 2.2. LA PLANIFICATION DES TÂCHES

Elle est de nature opportuniste : à l'aide d'une méthode d'essai-erreur, le superviseur essaie d'atteindre des buts ou des sous-buts qu'il a généré en fonction de la situation. Le moteur d'inférences fonctionne selon le cycle suivant :

(*a*) analyse de la situation sur des critères correspondant aux questions suivantes :

le but précédent est-il atteint ? (si non pourquoi ?)

où en est-on dans la phrase ?

quel est le problème à résoudre ?

quelles sont, parmi les informations disponibles, les plus sûres ?

quelles sont les opportunités (ensemble des tâches possibles) ?

*(les réponses à ces questions permettent de développer l'arborescence des actions possibles ou de revenir à un point de rendez-vous en cas d'impasse),*

(*b*) génération du plan d'action et du but à atteindre, ordonnancement des opportunités en fonction du but poursuivi et du risque estimé,

(*c*) exécution du plan (décomposé en sous-plans) par activation des experts,

(*d*) réception et organisation des informations dans le tableau noir,

(*e*) filtrage des hypothèses,

(*f*) gestion des points de rendez-vous,

(*g*) le but final est-il atteint ? oui : arrêt ; non : exécution d'un nouveau cycle.

Le raisonnement du superviseur sur lui-même se fait au début du cycle au niveau de la question « si non pourquoi ? ». En effet à travers la trace des actions effectuées et des buts qu'il fallait atteindre, il peut retrouver les causes de son échec et changer de stratégie.

### 2.3. LA BASE DE CONNAISSANCE DU SUPERVISEUR

Elle regroupe toutes les règles qui permettent l'analyse de la situation, le choix des opportunités et la gestion des hypothèses. Les plans activés sont fonction de la position dans la phrase et de la valeur prise par les drapeaux dans le tableau noir. Le tableau 1 précise les problèmes à traiter selon la hiérarchie des unités de décision mises en jeu au cours de la reconnaissance et la localisation courante du planificateur en reconnaissance. Ce sont ces problèmes à résoudre pas à pas qui guident la stratégie de reconnaissance : ils permettent de choisir les experts concernés et/ou les meilleures opportunités.

Les exemples de règles décrites ci-après donnent une idée des possibilités offertes par une telle méthode. Une stratégie possible — volontairement simplifiée — est donnée par la base de règles suivante :

**Initialisation « tableau noir » :**

début ;
Phrase_reconnue <- « faux » ; Parole <- « faux » ; Impasse <- « faux » ; Liaison <- « faux » ; Toutes les listes <- « vide » ;
fin ;

**Initialisation « base de faits » :**

début ;
mode <- proposition ; Liste-des-problèmes <- « vide » ;
ptr(pile(pr)) <- 0 ;
fin ;

**Méta-stratégie :** choix du type de stratégie selon l'état des variables d'environnement : si l'environnement n'est pas bruité on fait davantage confiance au DAP ascendant (Stratégie I développée ci-après), sinon si le vocabulaire n'est pas trop important on favorise une stratégie globalement descendante (Stratégie II non décrite ici)

début ;
si environnement-bruité = « faux » alors Stratégie I ;
sinon si taille-vocabulaire < 1 000 alors Stratégie II ;
sinon Stratégie III ;
finsi ;
finsi ;
fin ;

**Stratégie I :** stratégie de gauche à droite avec point de rendez-vous. L'appel fréquent au DAP ascendant confère à cette stratégie un caractère majoritairement ascendant. Elle n'est donc pas applicable que si le DAP a de bonnes performances

début ;
si Phrase-reconnue = « vrai » alors « but atteint » ;
stop ; finsi ;
si Parole = « faux » alors Plan « début-parole » ; Plan « début-syntagme » ;
sinon Plan « début-mot » ;
finsi ;
Recalage + ;
si Impasse = « faux » alors Plan « mot » ; Plan « fin-mot » ;
sinon « reconnaissance impossible » ; stop ;
finsi ;
Méta-stratégie
(le retour à la règle méta-stratégie permet ici de reconsidérer la stratégie à la fin de chaque mot)
fin ;

**Plan « début-parole » :** recherche du début de la parole par le DAP (cf. (1) tableau 1). Ce plan est dirigé par les données, une fois activé il n'est plus sous le contrôle du superviseur.

début ; DAP(p =) ;
si liste de phonèmes = non vide ET impasse = « faux »
alors parole = « vrai » ;
sinon Recalage + ; parole <- « faux » ; Plan « début-parole »
finsi ;
fin ;

TABLEAU 1

*Nature des problèmes en fonction de la localisation du système de reconnaissance au cours du traitement d'une phrase. Traitement signifie indifféremment proposition ou vérification.*

| *Localisation* / *unité* | début | dans | fin | entre |
|---|---|---|---|---|
| parole | (1) initialisation | | (2) conclusion | |
| phrase | (3) prédiction Synt/Sém | (4) traitement Synt/Sém | (5) vérification compréhension et prosodie | (6) traitement dialogue |
| syntagme | (7) prédiction Synt/Sém | (8) traitement Synt/Sém compréhension | (9) vérification Synt/Sém | (10) vérification prosodie |
| mot | (11) prédiction lexicale | (12) traitement phonétique | (13) filtrage lexical | (14) traitement des liaisons |
| phonème | (15) prédiction contexte | (16) traitement acoust/phon | (17) vérification phonétique | (18) vérification contexte |

**Plan « début-syntagme » :** prédiction de mots possibles par ASS et AL en début de phrase ou de syntagme (ce sont donc aussi des débuts de mots) (cf. (3) (7) tableau 1)

```
début ; liaison <- « faux » ; plan « début-mot » ; fin ;
```

**Plan « début-mot » :** fait les prédictions nécessaires en début de mot, compte tenu de la liaison précédente (cf. (11) tableau 1)

```
début ;
  si liaison = « vrai » alors S/Plan « filtrage » ;
  liaison <- « faux » ;
  Recalage + ; plan « début-mot » ;
    sinon ASS(p =) ; DAP(p =) ; AL(p =) ;
  finsi ;
fin ;
```

**Plan « mot » :** situation courante où la reconnaissance phonétique progresse à l'intérieur d'un mot (cf. (12) tableau 1)

```
début ;
  si mode = prédiction alors DAP(p =) ; AL(p =) ;
  AL(f =) ; finsi
  si mode = vérification alors AL(psc) ; AL(c =) ;
  DAP(V =) ; AL(f =) ; finsi ;
  si Liste de réussite = vide alors Plan « mot » ;
fin ;
```

**Plan « fin de mot » :** traite la fin de mots avec une liaison éventuelle. En cas d'impasse retour au point de rendez-vous précédent (cf. (13) (5) tableau 1)

```
début ;
  si Liste de réussite = non vide alors S/Plan
  « filtrage » ; C(v =) ;S/Plan « liaison-lat » ;
  S/Plan « liaison-phon » ; mode <- « proposition » ;
  impasse = « faux » ; Plan « fin-syntagme » ;
  finsi ;
  si impasse = « vrai » alors S/Plan « retour-arrière » ;
  S/Plan « élargissement-hps » ;
  finsi ;
fin ;
```

**Plan « fin-syntagme » :** traite la fin de syntagme et plus généralement la fin de phrase si celle-ci est attestée (cf. (9) (10) tableau 1)

```
début ;
  AP(p =) ; ASS(v =)
  si Liste de marqueurs prosodique = « pause » alors
  AP(f =) ; Recalage + ; finsi ;
fin ;
```

**S/Plan « filtrage » :** filtrage des hypothèses lexicales à la fin du développement d'un mot hypothèse

```
début ;
  si mode = prédiction alors AL(f =) ; finsi ;
  si mode = vérification alors AL(c =) ; DAP(v =) ;
  AL(f =) ; finsi ;
fin ;
```

**S/Plan « liaison-lat » :** prédiction des mots pouvant suivre un mot accepté, pour prédire une liaison latente (celle-ci est codée dans le lexique)

```
début ; ASS(p +) ; AL(p +) ; AL(p =) ;
  si liste-liaison = non-vide alors prédiction-liaison <-
  « vrai » ;
    sinon prédiction-liaison <- « faux » ;
  finsi ;
fin ;
```

**S/Plan « liaison-phon » :** test d'une liaison réalisée phonétiquement

```
début ;
  si prédiction-liaison = « vrai » alors DAP(p +) ;
  Ajout-phonème ; liaison <- « vrai » ;
    sinon liaison <- « faux » ;
  finsi ;
fin ;
```

**S/Plan « retour-arrière » :** l'ASS gère le retour-arrière et le contexte des hypothèses (il restaure les listes de réussite et d'échec obtenues au point de rendez-vous précédent)

```
début
  Recalage – ; ASS(pr –) ;
fin ;
```

**S/Plan « élargissement-hps » :** élargissement des hypothèses lexicales avant de repartir du dernier point de rendez-vous

```
début ;
  si Liste de réussite = non vide alors ;
    si prédiction-liaison = vrai alors
    S/Plan « liaison-lat »
    finsi ;
    AL(psc =) ; liaison <- faux ;
    impasse <- faux ; mode <- vérification ;
  finsi ;
fin ;
```

**Recalage + :** progression d'un pas vers la droite de la fenêtre courante. La commande « exit » permet de revenir directement à la règle méta-stratégie.

```
début ; t <- t + dt ;
  si fin-de-fichier alors
    si Liste réussite = « non vide » alors Phrase-recon-
    nue <- « vrai »
      sinon impasse <- « vrai » ;
    finsi ;
  exit ;
  finsi ;
fin ;
```

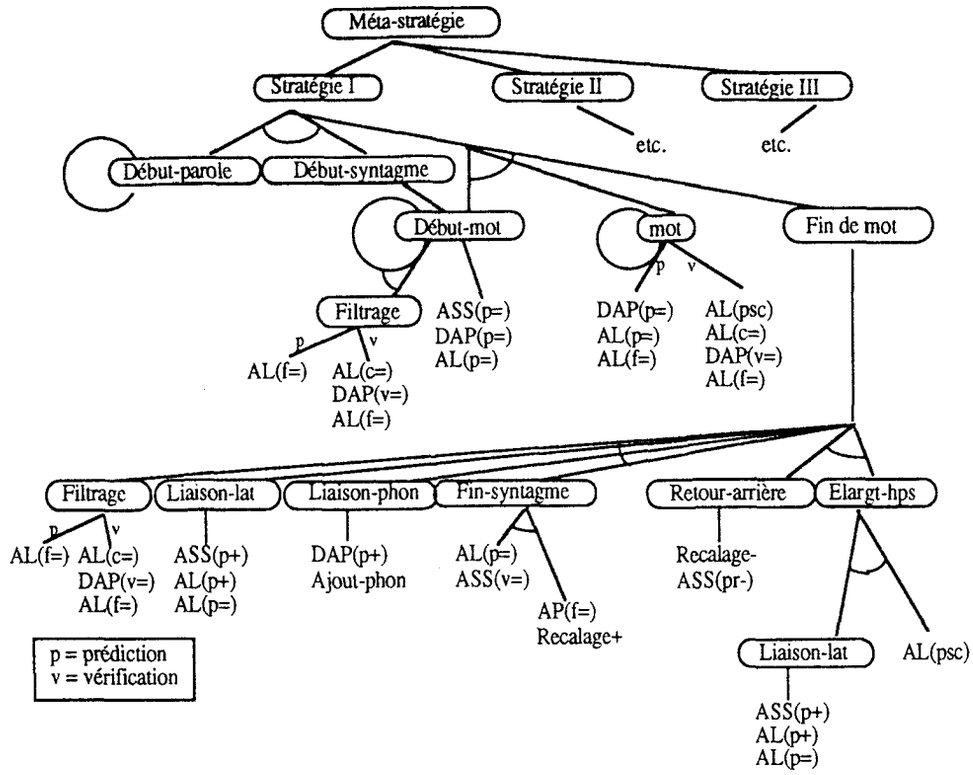
**Recalage – :** retour au point de rendez-vous précédent

```
début ;
  ptr(pile(pr)) <- ptr(pile(pr)) – 1
  si pile(pr) = « vide » alors impasse <- « vrai » ; finsi ;
fin ;
```

**Ajout-phonème :** ajout des phonèmes de liaison à la liste de phonèmes

```
début ;
  Liste de phonèmes <- Liste de phonèmes U Liste de
liaison ;
fin ;
```

Ce plan peut également être représenté par le graphe ET/OU suivant :




Cette stratégie **(Stratégie I)** utilise une recherche en faisceau avec retour en arrière. Elle est simplifiée dans la mesure où (*a*) elle n'utilise pas de coefficients de confiance pour mesurer les réponses des experts et (*b*) elle n'utilise pas de score pour mesurer le risque encouru à chaque étape du raisonnement : l'ordonnancement des tâches est statique. Cependant les résultats montrent (cf. ci-après) qu'elle est robuste surtout du fait de l'élargissement des hypothèses lors du retour-arrière : cet élargissement permet de trouver presque toujours une solution (mais au prix parfois d'une grande ambiguïté phrastique).

La base de faits du planificateur contient les valeurs par défaut pour initialiser le tableau noir ainsi que les variables d'environnement comme : « mode » = (proposition, vérification), « environnement-bruité » = (vrai, faux), « taille-du-vocabulaire » = entier, « type-de-syntaxe » = (S0, S1, S2), etc. A partir de là, il est facile de bâtir une méta-stratégie pour le système : (*a*) il faut d'une part, écrire et tester plusieurs stratégies pour des valeurs des variables d'environnement différentes puis (*b*) évaluer chaque stratégie sur des situations typiques et d'autre part (*c*) écrire les règles pour utiliser ces différentes bases en fonction des situations rencontrées. Pour le moment ce travail est effectué manuellement en attendant de disposer de mécanismes d'apprentissage semi-automatique voire automatique.

## 3. Le décodage acoustico-phonétique

Le rôle du DAP (Décodage Acoustico-Phonétique), fondamental en reconnaissance automatique de la parole, reste encore mal défini à l'heure actuelle. La difficulté du problème tient en partie au fait qu'il s'agit de projeter un sous-espace — celui des observations acoustiques assorties de toutes leurs variations — dans un espace plus vaste — celui des formes phonétiques — et pour une autre partie au fait que les niveaux linguistiques tout en étant hiérarchisés, sont fortement intégrés [35], [36], [51], [59].

Il y a donc un fossé à franchir — les ruptures substance/forme et continu/discret que l'on trouve entre les signaux acoustiques et les unités phonétiques. Ces ruptures interdisent de considérer le DAP comme une suite de transformations d'un domaine de représentation (continu) en un autre (discret) : en fait deux structures pré-existent et le DAP peut être considéré comme une mise en correspondance de la micro-structure acoustique du signal et de la macro-structure phonétique. Un premier travail est donc de décrire convenablement cette macro-structure dissimulée derrière les connaissances générales des sciences phonétiques. La mise en correspondance (« matching ») des modèles sous-jacents aux deux structures résout alors le problème de l'identification.

Dans une perspective qui utilise des connaissances explicites, plus précisément une technique de système-expert (ou plus largement d'IA) [43], [4], [6], [10], [24], il est plus efficace de définir les modèles résultant des macro-structures phonétiques sous forme de réseaux. Les transitions entre les états sont formulées à l'aide de connaissances et de règles idoines et le « matching » conduit à identifier ces macro-structures à partir de la micro-structure du signal (ou inversement) c'est-à-dire à parcourir convenablement les réseaux. Il en résulte un autre avantage qui réside dans la nécessaire séparation des connaissances liées à ces deux structures ; (*a*) les connaissances acoustiques qui restent très attachées aux procédures de calcul des paramètres (indices, corrélats, etc.) et (*b*) les connaissances phonétiques qui sont véhiculées essentiellement par les traits (informations symboliques pris au sens le plus large). La mise en correspondance peut alors être un processus tout à fait général et indépendant des connaissances mises en œuvre.

Le DAP revient donc soit à prédire la macro-structure à partir de la micro-structure soit à vérifier l'existence d'une micro-structure sous-jacente à une macro-structure donnée ; cela peut être schématisé de la manière suivante (*fig.* 2).

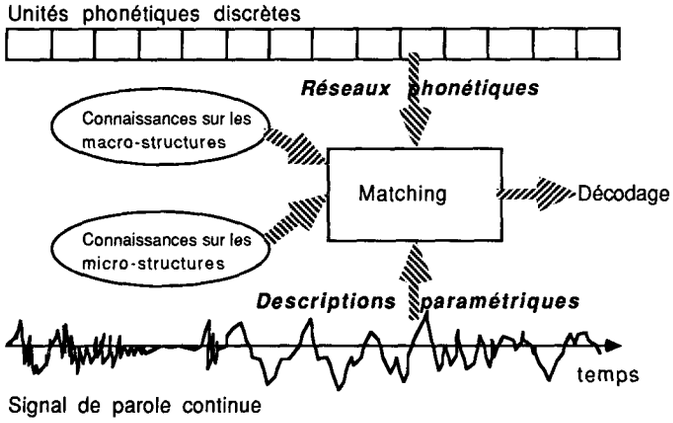



**Fig. 2. — Processus du décodage acoustico-phonétique : un modèle phonétique est donné a priori et sa structure est décrite dans les réseaux phonétiques ; d'un autre côté le signal a une description paramétrique connue. Des connaissances sur la macro-structure phonétique et la micro-structure acoustique sont rangées dans deux bases qui servent aux processus de mise en correspondance pour assurer l'identification. Le résultat produit peut être considéré comme le résultat de décodage.**

Les décodeurs acoustico-phonétiques fonctionnent en (*a*) proposition et en (*b*) vérification :

(*a*) le DAP **en proposition :** il délivre des informations aussi robustes que possible de type macro-trait puis, après filtrage et affinage, une liste de traits en fonction notamment du contexte courant,

(*b*) le DAP **en vérification :** il vérifie par « focalisation » la présence ou l'absence d'une information demandée par le superviseur — cette information peut être plus ou moins fine, comme phonème, trait ou macro-trait.

## 3.1. REPRÉSENTATION DES CONNAISSANCES

### 3.1.1. Les macro-classes

Classiquement les sons du français peuvent être divisés en macro-classes : les occlusives sourdes, les occlusives sonores, les fricatives sourdes, les fricatives sonores, les consonnes nasales, les consonnes liquides, les semi-voyelles, les voyelles et les pauses. Cette notion de macro-classe est souvent utilisée dans les systèmes utilisant un DAP pour en augmenter la robustesse [59], [56].

Certaines macro-classes ont de multiples façons de se réaliser. Par exemple, les semi-voyelles pourront être apparentées à des voyelles ou à des consonnes selon le contexte ou le locuteur, certaines fricatives sonores comme /*v*/ à des consonnes vocaliques. Il est donc préférable de regrouper certaines macro-classes pour augmenter d'une part les taux de reconnaissance pour la classification robuste et pour d'autre part, conserver la cohérence de la base de connaissance. Finalement cinq macro-classes ont été retenues : les occlusives sourdes, les fricatives (sourdes et sonores), les consonnes (vocaliques, nasales, liquides), les voyelles (dont les semi-voyelles co-produites), les pauses. A ces cinq macro-classes sont associés des réseaux phonétiques représentant la connaissance sur la macro-structure.

### 3.1.2. Les réseaux phonétiques

Un réseau phonétique $R_j$ est un réseau à une entrée et une sortie défini par le 5-uplet suivant :

$$R_j = \{j, S(j), T, s_{0j}, s_{fj}\}$$

avec

$j$ : identificateur du réseau,
S : ensemble des états,
T : ensemble des arcs (ou transitions $t_i$),
$s_0$ : état initial,
$s_{fj}$ : état final.

Les états représentent toutes les réalisations possibles des différentes phases acoustiques des macro-classes phonétiques, par exemple :

S(fricatives) : {début, friction-vocalique, cloison-début, friction-sourde, friction-sonore, closion-fin, fin} (*fig.* 4).

Une transition $t_i$ est définie par :

$$t_i = \{s_k, s_l, p_i, C_i, A_i\}$$

avec

$s_k$ et $s_l$ : extrémités de l'arc $t_i$,
$p_i$ : score attaché à la transition si celle-ci est parcourue,
$C_i$ : liste de contraintes devant être vérifiées lors du parcours de l'arc $t_i$,
$A_i$ : liste d'actions à effectuer en cas de succès.

Par convention le score global $p = \Sigma p_i = 0$ si les contraintes ne peuvent être vérifiées par le contrôleur de réseau lors d'une reconnaissance locale, sinon $p$ est la moyenne des scores $p_i$ pour toutes les transitions $i$ parcourues.

Les contraintes $C_i$ sont classées en trois catégories :

— les conditions de réalisation d'une phase acoustique (par exemple si l'énergie descend à un niveau suffisamment bas alors l'état « closion-fin » peut être atteint),

— les contraintes induites par le contexte (par exemple l'état « friction-vocalique » ne peut être atteint que si le phonème précédemment reconnu est une consonne vocalique ou un début de syntagme précédé d'une pause).

Les actions $A_i$ sont soit des procédures (calcul de paramètres, prédicats évaluables, etc.), soit des déroutements vers des règles à examiner de façon préférentielle — ce qui est en quelque sorte une forme de connaissance sur les contraintes articulatoires.

Contraintes et actions peuvent maintenant être mises sous forme de règles de production, les contraintes en constituant les prémisses et les actions, les conclusions. Mettre en correspondance la micro-structure et la macro-structure, revient donc à cheminer dans un réseau en parcourant les états de gauche à droite selon les règles actives à chaque pas. Ce cheminement est contrôlé par le mécanisme d'application des règles, ici celui de Prolog, langage dans lequel sont écrites les règles et qui convenait à ce problème.

A titre d'exemple nous décrivons ci-après le réseau des fricatives (*fig.* 3). Ce réseau contient 7 états représentant 5 phases acoustiques et 2 états fictifs :

début ; état d'entrée,
friction vocalique (début de la friction après une voyelle ou une consonne voisée),
début closion (petite chute d'intensité en début de friction),
closion fin (petite chute d'intensité en fin de friction),
friction sourde (friction sans voisement),
friction sonore (friction avec voisement),
fin : état de sortie.

Ce réseau montre la diversité des réalisations d'une fricative ou en d'autres termes la syntaxe des phases acoustiques : succession de plusieurs frictions sourdes ou sonores, ou closion suivie de frictions sourdes ou sonores, etc.

La règle qui contrôle les transitions vers le nœud « friction vocalique » est la suivante :

**SZ1-Règle** Friction_vocalique
*!transition Vocalique-fricative ou friction sonore à l'initiale*
**si** (Indice(Grave) > '+' OU (Fo ≠ 0 ET Crête_max < 1 000 Hz))
*!le phone candidat doit être « grave » ou présenter du voisement avec un formant en-dessous de* 1 000 Hz
ET bruitplusseuil < Energie < Bruit + (3/4)* Signal-sur-bruit
*!et être dans une fourchette d'énergie moyenne*
ET ∂(Aigu) ≥ '+'
*!puis devenir progressivement plus aigu*
ET (Etat(précédent) = Friction + Vocalique
*!la transition peut être une boucle*
OU (Contexte_antécédent = 'vocalique' ET Etat_précédent = 'néant')
*!le contexte précédent est soit vocalique*
OU (Contexte_antécédent = 'pause'
ET Etat(précédent) = 'néant'
*!soit un silence*

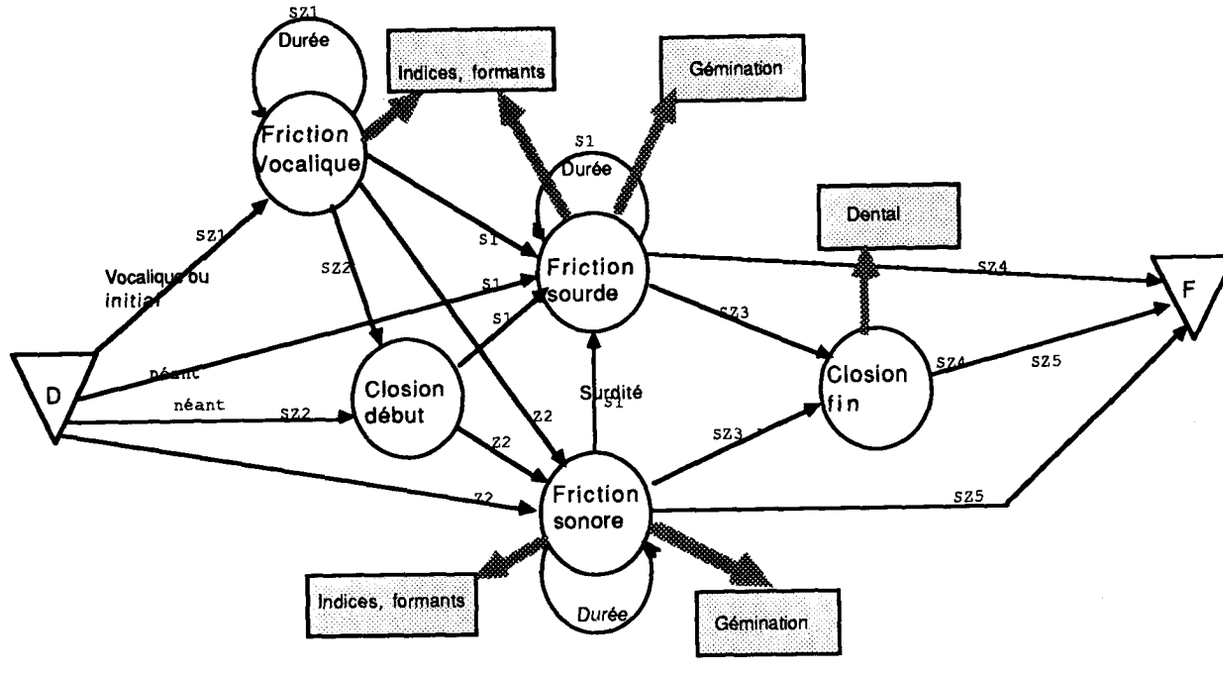



**Fig. 3. — Réseau des fricatives. Les notations $SZ_{ij}$, $Z_{ij}$ et $S_{ij}$ renvoient aux règles de transition pour les fricatives. Les états sont cerclés — sauf l'état de début et de fin —, les actions procédurales sont encadrées.**

ET pente(intensité) > $pente_1$))
!*il faut alors que dans ce cas l'intensité ait augmenté de façon significative*
**alors** Etat(courant) <- Friction_vocalique ; p <- 1
ET Action-proc(frontière_début <- phone_courant)
!*on mémorise les frontières*
ET Action-règle(SZ1 OU SZ2 OU S1 OU Z2)
!*on active ces règles pour cheminer dans les transitions suivantes*

On reconnaît sur cette règle les contraintes des deux types et les actions décrites ci-dessus.

### 3.2. STRATÉGIES DE DÉCODAGE

L'architecture du module de décodage acoustico-phonétique, utilisant ces connaissances est structurée autour des modules suivants (*fig.* 4) :

— Un module d'analyse acoustique qui paramétrise le signal en créant toutes les 8 ms un spectre défini sur 24 canaux à partir d'un modèle d'oreille [7] sur lequel sont calculés 6 indices acoustiques (grave/aigu, fermé/ouvert, écarté/compact, bémolisé/diésé, doux/strident, continu/discontinu) [8]. Le signal est ensuite segmenté en phones homogènes au vu des discontinuités des indices et de l'énergie [53]. Parallèlement, le fondamental est calculé par une méthode AMDF améliorée [1].

— Deux modules de décodage : le DAP($p$) propose des informations sûres à partir du signal. Ces informations doivent être les plus robustes possible : pour cela elles sont hiérarchisées en macro-traits et traits de manière à ce que le DAP ne soit pas tenu de fournir des traits si le coefficient de certitude obtenu est insuffisant. Le DAP($v$) vérifie une information phonétique a priori (macro-trait, trait ou phonème) sur le signal.

### 3.3. RÉSULTATS

Pour le DAP($p$) le critère principal de qualité est sa robustesse, c'est-à-dire sa capacité à transmettre des informations fiables même si celles-ci sont insuffisantes pour la caractérisation non ambiguë des phonèmes. Or cette robustesse dépend beaucoup des performances obtenues pendant l'étape de localisation : en effet autant il est facile de rétablir un trait — par vérification descendante après un contrôle linguistique par exemple — ou de filtrer un candidat redondant — par des règles phonotactiques par exemple — autant il est pratiquement impossible de remettre en question dans une étape linguistique, une décision sur la localisation — il faudrait par exemple posséder pour cela de paramètres articulatoires explicites et de relations acoustico-articulatoires claires. Aussi, devant l'importance de ce problème et dans le cadre restreint de cet article nous nous contenterons de commen-

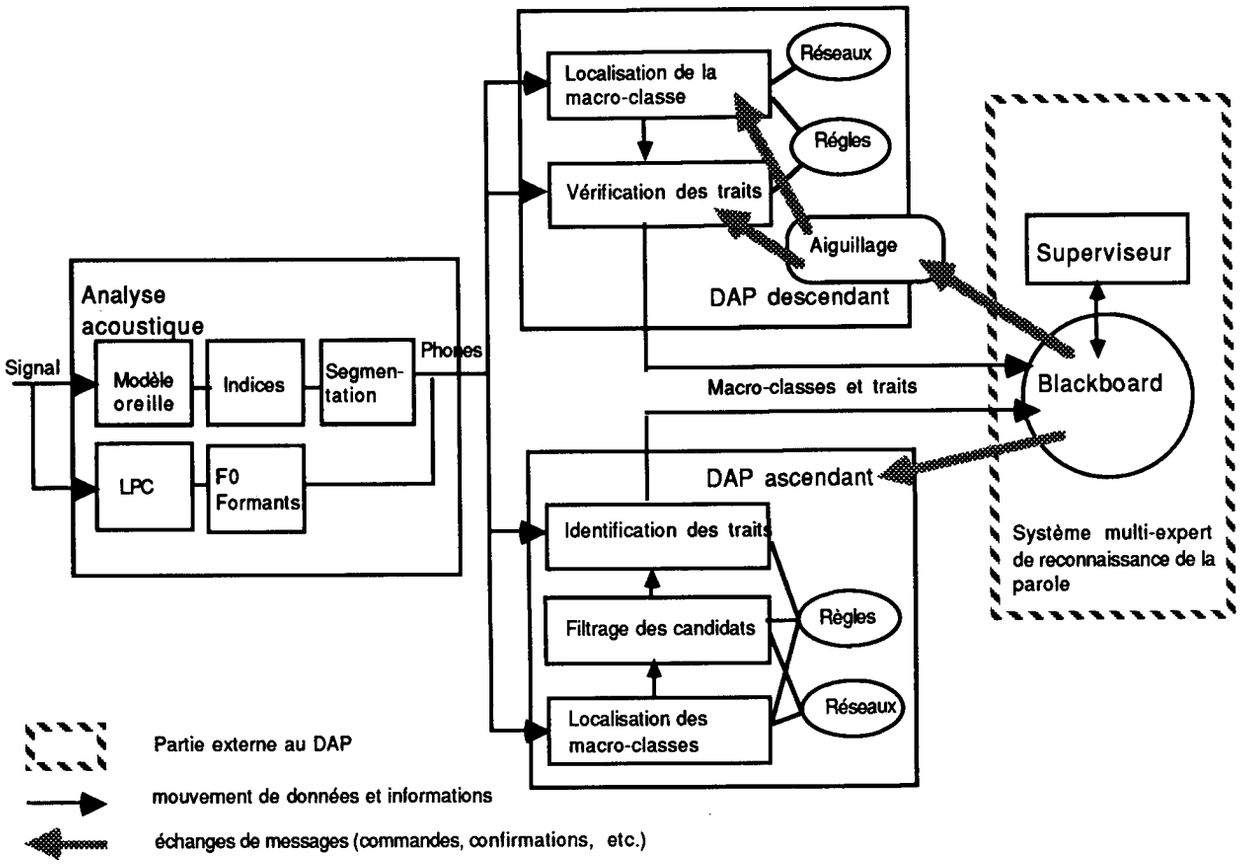



**Fig. 4. — Schéma général du Décodage Acoustico-Phonétique, les experts DAP($p$) et DAP($v$).**

ter les résultats concernant la phase de localisation et de reconnaissance des macro-classes (ou macro-traits).

Sans apprentissage, sur un corpus d'évaluation de BDSONS (Base de Données des Sons du Français) — les phrases phonétiquement équilibrées de deux locuteurs homme et femme — les principales sources d'erreur pour la localisation proviennent des consonnes liquides. Les consonnes nasales sont parfois étiquetées « voyelle » car le phonème /*m*/ — très énergétique pour ces locuteurs — est souvent confondu avec une voyelle, par contre le phonème /*n*/ est bien détecté comme consonne. Les résultats montrent que les phases acoustiques trouvées par le système de reconnaissance sont bien localisées et bien étiquetées ($R_{moyen}$ = 88,6 % sur les 1 177 phonèmes du locuteur masculin et de 92,5 % si les consonnes liquides ne sont pas prises en compte, $R_{moyen}$ = 85,8 % sur les 923 phonèmes du locuteur féminin et de 93,2 % si les consonnes liquides ne sont pas prises en compte).

Les résultats obtenus lors de la phase de localisation du DAP(*p*) montrent que la base de connaissance est robuste (et multilocuteur) car les réseaux décrivent bien la structure temporelle des sons, les erreurs venant essentiellement des consonnes liquides (à 50 %) très influençables par les contextes — donc sans forme spécifique — et par 15 % des consonnes nasales très énergétiques.

En conclusion, la structuration des connaissances sous forme de réseaux phonétiques permet de séparer clairement les connaissances sur la macro-structure phonétique et sur la micro-structure du signal de parole et d'aborder la reconnaissance des unités phonétiques à travers la syntaxe de leurs phases acoustiques. La notion de réseau phonétique permet aussi de jeter un pont entre les modèles stochastiques tels que HMM — dans lesquels la macro-structure (sur les axes paradigmatiques et syntagmatiques) de la parole n'émerge pas suffisamment — et les systèmes à règles de production. Cela permet donc d'introduire un niveau de contrôle à partir de connaissances phonétiques explicites, pour des systèmes mixtes.

# 4. Les analyseurs linguistiques

On sait [58] que les ATN (Augmented Transition Networks) sont les outils parmi les plus puissants pour résoudre les problèmes posés par la représentation et la mise en œuvre des connaissances syntaxico-sémantiques [50]. En effet ils supportent diverses caractérisations syntaxiques et s'accommodent très bien des contraintes sémantiques. Par ailleurs, le système DIRA utilise déjà une technique ATN (réseaux phonétiques) pour le niveau acoustico-phonétique (cf. § 3). Il est donc tentant d'uniformiser toutes ces approches « réseaux » avec un même formalisme pour donner une cohérence maximale au système de reconnaissance.

Pour ces raisons, l'analyseur syntaxico-sémantique est modélisé à l'aide d'ATN [48] et prend appui directement sur le lexique. On sait cependant que la mise en œuvre de ces ATN n'est pas très commode pour un utilisateur non averti ; c'est pourquoi, pour faciliter l'écriture de la grammaire et du lexique, un compilateur a été développé ; à partir d'une représentation externe, il produit un ATN en représentation interne muni de relations avec le lexique, lorsqu'on lui fournit une grammaire quelconque en entrée. La question reste cependant de savoir si une telle représentation peut s'accommoder des processus cognitifs humains [16], [55], [3], [15], [22], [28], etc.

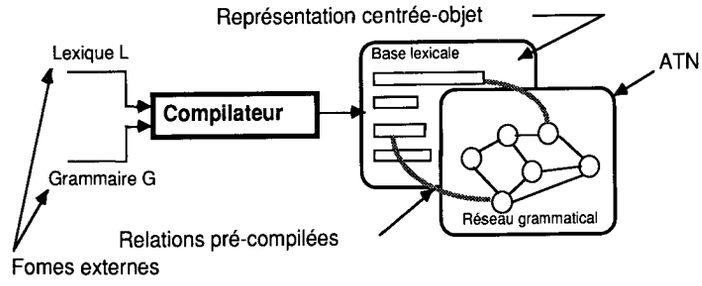



**Fig. 5. — Le compilateur d'ATN lexico-syntaxico-sémantique.**

Dans la représentation externe, les règles classiques de réécriture sont assorties de deux champs supplémentaires : un champ « contexte » et un champ « actions ».

## 4.1. CONTRAINTES SUR LES ACCÈS LEXICAUX

L'analyse syntaxico-sémantique est toujours liée au lexique selon les deux perspectives suivantes :

— la vérification — où il s'agit de confirmer ou d'infirmer qu'une suite de mots est syntaxiquement correcte — exige un accès au lexique pour rechercher les attributs syntacti-co-sémantiques des mots à vérifier, ces mots étant connus par leur transcription phonétique,

— la prédiction (très importante pour le traitement de la parole) — où il s'agit de fournir une liste de candidats-mots possibles après une séquence correcte — exige aussi un accès au lexique à partir des attributs syntaxico-sémantiques prédits par l'analyseur syntaxique (en examinant cette fois tous les chemins possibles dans l'ATN à partir d'un état origine donné).

Dans tous les cas, il est évident que la relation syntaxe-sémantique-lexique est très forte et doit être prévue au moment de la compilation de l'ATN afin de diminuer, entre autres, le temps de la recherche (une des techniques est de prévoir par avance un accès « statique » précompilé). Cette relation est prise en compte, sous forme de contrainte d'accès, directement dans le lexique — elle porte sur les catégories syntaxiques, sémantiques, attributs, etc. Il est évident que des « actions » placées en partie droite de règles — comme il est de coutume dans les ATN — pourraient résoudre le même problème, mais le temps d'exécution serait plus long puisque les accès seraient calculés à chaque fois : nous avons préféré les matérialiser par des pointeurs donnant un accès direct aux items lexicaux. Ces contraintes d'accès sont indiquées, si

nécessaire, explicitement, dans les règles immédiatement après chaque terme.

Exemple de règle en langage externe :

S1 : SN -> Dét N/accprd($1, $2),/ ; /*action d'accord entre le $1 = Dét et le $2 = N*/

S2 : SN -> Dét N Adj($qual) ; /*accès restreint aux adjectifs qualificatifs seulement*/

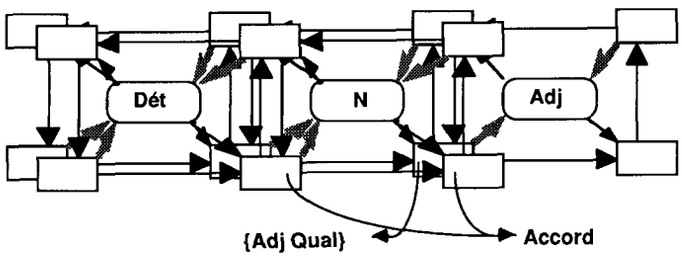



**Fig. 6. — Réseau ATN compilé pour les deux règles S1 et S2. Les chaînages entre nœuds sont prévus pour permettre une analyse de droite à gauche et inversement. Un pointeur spécifique est créé pour accéder aux adjectifs qualificatifs (Adj Qual) dans le lexique et une action (Accord) est instanciée à la fois au niveau (Dét N) et au niveau (N Adj).**

### 4.2. L'ANALYSE SYNTAXICO-SÉMANTIQUE

Une analyse va consister à parcourir le réseau selon le type de fonctionnement fixé par le superviseur. Cette analyse est effectuée par le contrôleur de réseau appelé « analyseur ». Deux modes d'analyse sont prévus : le mode ascendant et le mode descendant. Pour chaque mode et à tout moment de l'analyse, deux fonctionnements sont possibles : (*a*) un fonctionnement en vérification et (*b*) un fonctionnement en prédiction. Pour vérifier une chaîne d'entrée, l'analyseur cherche un chemin dans le réseau à partir du nœud courant. En prédiction, l'analyseur propose tous les nœuds possibles, successeurs à la distance $k$, du nœud courant.

A partir de points d'ancrage syntaxiques, comme les débuts ou fins de phrase, de syntagme, l'analyse descendante est bien appropriée. Par contre, si l'analyseur ne connaît pas la position syntaxique courante mais s'il connaît un point d'ancrage du niveau de description du vocabulaire terminal, l'analyse ascendante sera activée. L'analyseur autorise les sens de parcours gauche-droite et droite-gauche. Ainsi, le superviseur pourra activer une analyse du milieu vers les côtés en partant des points d'ancrage.

L'analyseur permet la gestion des règles récursives. Il maintient deux piles, l'une pour les règles à contexte libre et l'autre pour les règles transformationnelles ou contextuelles. Il construit en parallèle toutes les solutions syntaxiquement et sémantiquement correctes. En fin d'analyse, il fournit un arbre de solutions syntaxiquement et sémantiquement correctes, la structure des constituants (*c*-structure), ainsi qu'une liste de solutions fonctionnelles (*f*-structure).

## 5. La compréhension dans le système DIRA

Le concept même de compréhension reste variable selon les auteurs : pour nous il s'agit d'une phase d'interprétation qui doit conduire à une représentation exécutable par la machine. Mais, pour rester compatible avec l'ensemble de la stratégie choisie pour le système DIRA, la compréhension doit se faire au fur et à mesure de la reconnaissance, c'est-à-dire dans le sens gauche-droite, tantôt en vérification tantôt en prédiction. La vérification doit se faire sur des séquences de mots — phrases incomplètes — présentes dans le tableau noir à un instant donné, dans le but de (*a*) filtrer les hypothèses en cours et (*b*) débloquer la suite des traitements. La prédiction doit permettre de faire des hypothèses sur quelques structures de phrases et mots possibles pour faciliter la prédiction lexicale. Dans les deux cas nous avons choisi de nous appuyer sur le concept d'*amorçage sémantique*.

Schématiquement le processus d'amorçage sémantique (appelé aussi amorçage lexical par certains auteurs) est un processus cognitif dans lequel un mot en appelle d'autres par association de sens [2], [41], [33]. Ces mots peuvent être considérés comme appartenant à un même champ sémantique. Ils sont parfois à des distances syntagmatiques lointaines les uns des autres dans le discours pour lesquels, donc, il semble que la syntaxe ne joue pas. L'amorçage sémantique peut fonctionner aussi bien dans le sens direct (forward priming) que rétrograde (backward priming) [25]. Ce phénomène est à la fois lié et à la fois indépendant de la syntaxe : les relations entre les mots des constituants mineurs (ex. morceau de musique) et des constituants majeurs ne se comportent pas de la même manière. Dans le premier cas une syntaxe incorrecte bloque le processus (ex. musique de morceau), dans le second cas elle peut le faciliter (ex. *donner livre* à Jean). Ce phénomène est probablement assez précoce dans les processus de traitements linguistiques humains et s'insère avant la prise en compte des informations pragmatiques.

Tout ceci nous a donné l'idée d'introduire une telle composante entre l'analyse syntaxico-sémantique et la pragmatique dans le système DIRA.

### 5.1. LA COMPOSANTE PRAGMATIQUE

Nous ne détaillerons pas cette composante qui présente toutes les caractéristiques habituelles que l'on rencontre dans les sytèmes de dialogue — à savoir les bases de connaissances statiques comme objets de l'univers, tâches, et les connaissances dynamiques comme les historiques — et qui fonctionne à l'aide d'une grammaire de cas [21].

### 5.2. LA COMPOSANTE SYNTAXICO-SÉMANTIQUE

Cette composante délivre la *c*-structure et la *f*-structure conformément au modèle de LFGrammar (Grammaire Lexicale Fonctionnelle) de Bresnan et Kaplan [32], [5] mis en œuvre dans l'ATN décrit, ci-dessus.

La description interne d'une phrase est formée de deux éléments indépendants :

1. La structure des constituants ou *c*-structure pour les aspects syntaxiques,

2. La structure fonctionnelle ou *f*-structure pour les aspects sémantiques.

Dans le modèle LFG, la *c*-structure est obtenue par une analyse classique de la phrase avec une grammaire indépendante du contexte [14]. Cette grammaire est censée couvrir toutes les formes possibles de phrases, elle est beaucoup plus large qu'une grammaire syntagmatique seule puisqu'elle autorise des structures incorrectes qui sont filtrées ultérieurement par la *f*-structure. Cette dernière est fondée sur la notion de schéma. Elle est engendrée à partir des équations qui sont associées aux règles de la grammaire.

Par convention on considère que :

$p$ : désigne la structure fonctionnelle du nœud père,
$f$ : désigne la structure fonctionnelle du nœud fils.

Par exemple, soit la règle :

$$\begin{array}{lll} P \rightarrow & GN & GV \\ & p \text{ sujet} = f & p = f \end{array}$$

Ici $p$ désigne la phrase P, et $f$ désigne successivement GN, GV. Cette équation signifie que le GN est sujet de P et de GV et que la structure fonctionnelle de la phrase est portée par la structure fonctionnelle du groupe verbal GV.

La production (ou l'analyse) d'une phrase suit donc trois étapes :

1. On produit un arbre de dérivation dont les feuilles sont toutes des catégories lexicales en utilisant la grammaire non contextuelle, et en négligeant les équations qui sont attachées aux règles.

2. On complète chaque feuille par un mot approprié du dictionnaire, puis selon les dérivations et les mots choisis, on met à jour les équations (en remplaçant les variables $(p, f)$ qui y figurent par les termes convenables).

3. Ces équations, jointes à celles qui sont attachées aux entrées du lexique choisies, sont résolues et fournissent la structure fonctionnelle de la phrase.

### 5.3. LA COMPOSANTE CONSTITUANTS MAJEURS ET CONSTITUANTS MINEURS

Son rôle est de traiter de l'amorçage sémantique comme indiqué ci-dessus :

(*a*) en vérification, elle établit les relations, de type « forward » et « backward », entre certains mots des constituants de la phrase, et écrit dans la mémoire commune un drapeau qui indique le degré de compréhension atteint,

(*b*) en proposition, elle prédit les structures de sens qui peuvent convenir après une séquence de mots donnés.

Pour le moment seule la partie (*a*) a été implantée dans le système de la manière suivante :

Les relations entre les constituants sont classées dans des groupes de relations qui dépendent de la structure fonctionnelle de la phrase. Ceci est illustré ci-après pour quelques cas :

1. les phrases de type (GV) :

$$\begin{array}{l} P \rightarrow GV \\ \quad p = f \end{array}$$

L'analyse fonctionnelle de la phrase se ramène ici à celui du groupe verbal. Il n'y a qu'un seul constituant majeur (CM) et donc aucune vérification entre CMs à effectuer. Par contre à l'intérieur du GV il peut y avoir plusieurs constituants mineurs (Cm) comme V Adv. La cohérence entre V et Adv doit être assurée (« avancer chaudement » n'a pas de sens *en soi* et l'on voit qu'il n'est pas besoin d'attendre le traitement pragmatique pour éliminer cette solution). Le schéma associé au verbe est au moment de l'analyse :

*verbe :* *item = avancer*
*temps = infinitif*
*modulateur = adverbe_de_mouvement*
*concept = <verbe,* COD = *vide,* COI = *vide >*

avec COD = complément objet direct ; COI = complément objet indirect.
Il suffit de vérifier que « chaudement » n'est pas un adverbe de mouvement.

2. Les phrases de type (GV GN) :

$$\begin{array}{ll} P \rightarrow GV & GN \\ \quad p = f & p \text{ COD} = f \end{array}$$

L'analyse fonctionnelle délivre GN = COD(GV) et l'analyse en constituants deux CM. Le problème est donc ici d'évaluer la relation de type forward entre le verbe du GV et le nom du GN. Donc il faut établir une relation logique du type (verbe, nom) à travers les schémas du verbe et du nom pour en établir la vérité.

3. Les phrases de type (GV GP) :

$$\begin{array}{ll} P \rightarrow GV & GP \\ \quad p = f & p \text{ COI} = f \\ GP \rightarrow prép & GN1 \\ & p = f \end{array}$$

Les verbes dans ce cas (ici pour le verbe « avancer« ) sont représentés par :

*verbe :* *item = avancer*
*temps = infinitif*
*modulateur = adverbe_de_mouvement*
*concept = <verbe,* COD = *vide,* COI = *destination>*

Pour comprendre ce type de phrase il faut mesurer l'amorçage sémantique du type « forward » entre le verbe du GV et la préposition du GP, puis l'amorçage de type « forward » entre le verbe du GV et le nom du GN1 et enfin l'amorçage entre les prépositions du GP et le nom du GN1 c'est-à-dire les relations logiques de type (verbe, prép) et (verbe, nom) et (prép, nom).

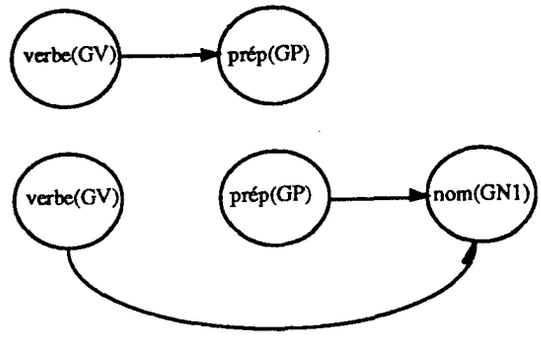



**Fig. 7. — L'amorçage sémantique « forward » dans la phrase de type (GV GP).**

Ex. : la phrase « aller vers la porte » devient compréhensible si l'on arrive à établir les relations entre (aller, vers), ET (aller, porte) ET (vers, porte).

4. etc. pour tous les autres types de phrase possibles.

### 5.4. LA STRATÉGIE DE COMPRÉHENSION

La compréhension d'une phrase est donc soumise à deux conditions :

(*a*) elle doit être syntaxiquement correcte localement, et,

(*b*) l'amorçage sémantique entre les mots lexicaux concernés de la *c*-structure doit pouvoir s'établir.

La stratégie consiste donc à activer ASS($v$ = ) en mode descendant de gauche à droite pour vérifier tout d'abord que les structures sont correctes. Dans ce cas l'ASS envoie un message au superviseur pour lui indiquer d'activer C($v$ = ) dont le rôle est alors de vérifier l'amorçage sémantique entre les constituants déterminés par l'ASS et présents dans le tableau noir. En cas de succès, la phrase (ou portion de phrase) est dite « compréhensible » ce qui permet au superviseur de valider ses hypothèses et de poursuivre le plan.

## 6. L'Analyse prosodique

On accorde à la prosodie beaucoup de potentialités dans un système de reconnaissance : tant au niveau de la microprosodie pour séparer les consonnes des voyelles, que des accents de mots ou de phrases, ou même de l'expression stylistique. Jusqu'à présent peu de systèmes intègrent la prosodie tant il est vrai que son rôle reste difficile à cerner. Nous avons tenté, à travers une première étude, de donner un rôle précis à la prosodie dans le système DIRA.

Le contour prosodique d'une phrase dépend de nombreux facteurs parmi lesquels :

(*a*) le contexte linguistique qui influe sur le style d'élocution,
(*b*) le type de communication (discours naturel, dialogue, lecture, etc.),
(*c*) le contexte textuel à travers les modalités de phrase — énonciatif/interrogatif — ou la structuration de l'énoncé,
(*d*) la variabilité inter-locuteur (facteurs socio-linguistiques, débit d'élocution),
(*e*) la stratégie utilisée dans la communication (répartition des accents, découpe en mots, etc.),
(*f*) la situation pragmatique incluant le(s) destinataire(s) du discours.

Le contour prosodique est donc très variable d'un locuteur à l'autre, pour une même situation de dialogue. Néanmoins il se réfère au système linguistique de la langue et c'est pourquoi il représente un champ d'investigation intéressant la reconnaissance automatique de la parole [40].

La prosodie apporte en effet, des informations à divers niveaux dans le processus de la reconnaissance :

— *phonétiques,* à travers la microprosodie (on sait que les voyelles ouvertes sont plus énergétiques que les voyelles fermées ou que la micromélodie des occlusives sonores est différente de celle des consonnes nasales ou que le VOT (Voice Onset Time) est un indice intéressant pour les occlusives, etc.),

— *lexicales,* les accents de mots ne peuvent pas être placés aléatoirement, ils dépendent en premier lieu de la position du mot dans le groupe prosodique ainsi que du nombre de syllabes,

— *syntaxico-sémantiques,* les marqueurs sémantiques (culmination dans le processus de la focalisation du sens par exemple) ou syntaxiques (distribution des pauses, allongement de syllabes) contribuent à la ponctuation orale de la phrase sur le plan de sa structure.

Il serait donc utile de repérer automatiquement ces marqueurs en reconnaissance [12], soit (*a*) pour prédire la présence de frontières de mots ou de groupes de mots en vue de contraindre la reconnaissance, notamment les accès lexicaux, soit (*b*) pour vérifier l'adéquation d'une hypothèse de structure de phrase (en termes de rythme, frontières de syntagme, frontières de mots) par rapport au contour prosodique.

Plusieurs questions se posent au sujet de la prosodie et de son utilisation en reconnaissance :

1. Existe-t-il des schémas prosodiques « profonds » utiles à la reconnaissance et qu'un traitement adéquat des paramètres prosodiques pourrait faire apparaître ?

2. Comment doit-on utiliser les informations prosodiques disponibles dans une stratégie ascendante (bottom-up) et descendante (top-down) ? Ou en d'autres termes doit-il y avoir plusieurs stratégies au sujet de la prosodie ?

3. Comment faire coopérer les sources de connaissances prosodiques avec les autres sources de connaissances dans un système de reconnaissance ? [39]

### 6.1. LES PARAMÈTRES PROSODIQUES

Pour utiliser les informations prosodiques, il convient de disposer de paramètres locaux (à court terme) et globaux (squelettes de contours à long terme) c'est-à-dire des informations caractéristiques de la structure de surface et de la structure profonde, voire d'une structure intermédiaire.

Pour la structure profonde il y a lieu en fait de distinguer une structure de phrase et une structure de texte : ce deuxième cas n'est pas envisagé dans cet article pour des textes très longs (les valeurs prosodiques retenues sont en effet normalisées phrase par phrase).

### 6.1.1. Détection des noyaux vocaliques

La structure prosodique de la phrase en français [49], [11], [52] peut-être dégagée à l'aide des trois paramètres $F_0$ (pitch), E (intensité ou énergie) et D (durée) calculés sur les *syllabes*. Mais la localisation des syllabes en analyse ascendante est une gageure : du fait qu'elles dépendent plus du niveau phonologique et lexical que du niveau acoustique, leur début et leur fin ne sont pas toujours clairement repérables et donc la mesure précise de leur durée n'est pas envisageable. Sur la courbe mélodique, l'instant de prélèvement des valeurs de $F_0$ est également sujet à discussion : par exemple, aux 2/3 de la voyelle [49] ou à l'instant de maximum de stabilité ? En ce qui concerne la courbe d'intensité des problèmes de mesure se posent également : (*a*) l'échelle (*d*B, linéaire, pondération perceptive, etc.), (*b*) la bande passante, (*c*) la fenêtre temporelle.

Devant tous ces niveaux de difficulté, une stratégie raisonnable consiste à localiser les centres des voyelles ou cibles atteintes — qui émergent suffisamment bien sur la courbe d'intensité — puis à s'étendre de part et d'autre pour obtenir une zone de stabilité suffisante pour le calcul des paramètres prosodiques moyens. Nous appelons ces zones « noyaux vocaliques » — en effet elles correspondent souvent à des émergences vocaliques, soit des voyelles soit des consonnes énergétiques. Pour cela, sur le plan acoustique il est possible de détecter des ruptures sur les courbes prosodiques calculées pour chaque trame du signal. Ces ruptures ne correspondent pas toujours à des frontières de syllabes mais le segment le plus intense compris entre deux ruptures peut être assimilé à un noyau vocalique (*fig.* 8). Dans ces conditions, en regroupant deux segments successifs « intense » et « non intense » on peut définir une unité prosodique proche de la syllabe pour laquelle la localisation acoustique ne pose pas de problème majeur : nous appelons cette unité une « pseudo-syllabe ». Seulement, le problème est de savoir si ces pseudo-syllabes ont un sens et si elles peuvent être utiles vis-à-vis de la fonction démarcative de la prosodie.

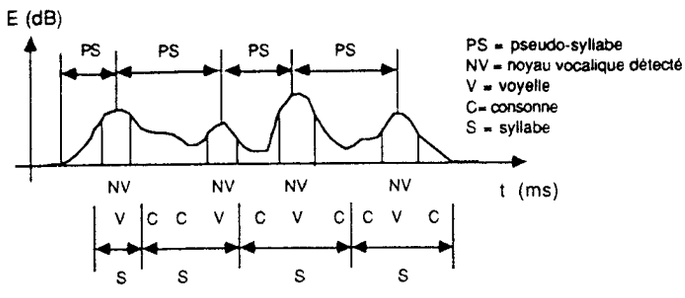



**Fig. 8. — Schématisation des notions de noyau vocalique (NV) et de pseudo-syllabe (PS). On remarque que la notion de pseudo-syllabe peut s'éloigner notablement de la notion de syllabe (S) en particulier lorsque la structure syllabique devient complexe (CCV, CVC, VCC, etc.). En fait la durée de cette pseudo-syllabe est fortement corrélée au rythme des voyelles dans la phrase.**

Le taux de détection global de ces noyaux vocaliques est de 98,5 % (nombre de noyaux correctement détectés/nombre de noyaux étiquetés) sur les phrases phonétiquement équilibrées de BDSON pour trois locuteurs (2 hommes et 1 femme, parole lue). Un noyau vocalique est déclaré bien détecté lorsque l'étiquette de voyelle (posée par un expert au centre du phonème) est à l'intérieur du segment correspondant à ce noyau vocalique.

Les erreurs de détection des noyaux vocaliques peuvent être rangées en deux catégories :

(*a*) cas de sous-segmentation

succession de deux voyelles V/V (40 % du total des erreurs)

(*b*) cas de sur-segmentation

succession semi-voyelle voyelle Sv/V (30 %)
les voyelles longues en fin de syntagme (10 %)
les voyelles longues en finale ayant une énergie basse (20 %)

Les conséquences des erreurs de détection dans les séquences CV, VV, CCV est moindre qu'il n'y paraît : en effet cette détection n'est faite que pour préparer la phase de squelettisation des courbes prosodiques et les erreurs ne sont pas cumulables avec celles du niveau suivant. On remarque que les erreurs relatives les plus importantes concernent la sous-segmentation V/V c'est-à-dire le cas où un seul noyau est détecté au lieu de deux : dans ce cas ce noyau est souvent plus long et risque de ce fait de porter un accent de durée. Par contre dans le cas de sur-segmentation d'une semi-voyelle/voyelle en deux noyaux, on peut assimiler ce problème à une diphtongue — bien qu'il n'en existe pas en français — de manière à rattraper a posteriori ce type d'erreur. Dans les deux derniers cas de sur-segmentation des voyelles longues, cela n'est pas nuisible car les deux noyaux détectés restent encore suffisamment longs pour garder une marque d'allongement.

### 6.1.2. Le squelette prosodique

Les paramètres prosodiques calculés sur les noyaux vocaliques sont :

$NV(n)$ = noyau vocalique n° $n$,
$E_c(n)$ = énergie codée sur 4 niveaux,
$F_0(n)$ = valeur de $F_0$ sur $NV(n)$,
$F_c(n)$ = fréquence fondamentale $F_0(n)$ codée sur 4 niveaux,
$D(n)$ = durée entre les deux noyaux vocaliques $NV(n)$ et $NV(n-1)$,
$\partial D(n)$ = $D(n) - D(n-1)$ ($\partial D \langle 0 / \partial D \rangle 0 \Rightarrow$ accélération/ralentissement),
$\partial D_c(n)$ = $\partial D(n)$ codée sur 16 niveaux localement dans la phrase,
$DLD(n)$ = différence entre la ligne de déclinaison de la durée et la durée $D(n)$,
$DLD_c(n)$ = $DLD(n)$ codée sur 8 niveaux,
$DLF_c(n)$ = différence entre la ligne de déclinaison de $F_0$ et la valeur locale $F_0(n)$.

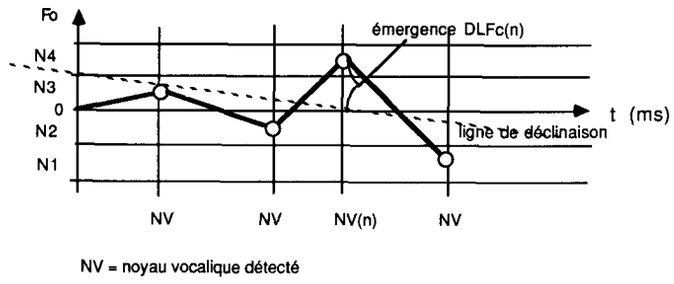



**Fig. 9. — La ligne de déclinaison sur un paramètre est calculée par ajustement linéaire sur les valeurs de ce paramètre sur les noyaux vocaliques successifs. Les valeurs d'émergence telles que DLFc s'en déduisent par différence. Les niveaux N1 à N4 sont normalisés d'une phrase à la suivante et répartis linéairement sur le support de variation du paramètre. Les notions de niveaux sont inspirées de [17].**

La visualisation de ces paramètres sous forme de courbes squelettisées conduit à la figure 10 :

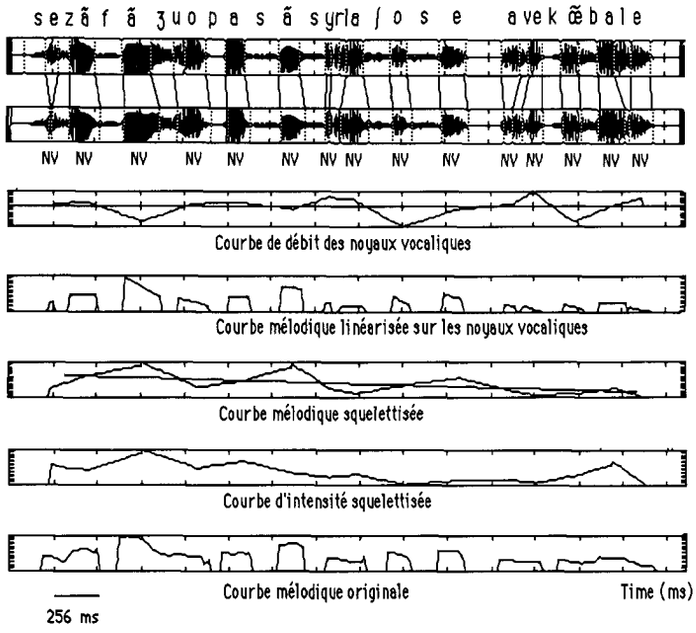



**Fig. 10. — Paramètres prosodiques et noyaux vocaliques de la phrase « Ces enfants jouent aux passants sur la chaussée avec un balai » (phrase tirée d'un corpus de V. Aubergé). De haut en bas : le signal et les frontières phonémiques détectées par un phonéticien, les noyaux vocaliques calculés, les courbes prosodiques lissées.**

### 6.1.3. Règles prosodiques

La base de règles issue des connaissances d'un expert (G. Caelen-Haumont) et mise en œuvre dans l'analyseur prosodique [47] a pour but de positionner les éléments principaux de la phrase afin d'appuyer les niveaux linguistiques dans le processus de prédiction et de vérification. Les marqueurs prosodiques sont les suivants :

DP = début_de_phrase
FP = fin_de_phrase
AI = accent_d'intensité
AM = accent_mélodique
MG = mot_grammatical
DM = début_mot_lexical
FM = fin_mot_lexical

Les règles s'écrivent simplement sous la forme suivante :

(*a*) détection de fin de phrase affirmative

Ph1. *SI* $(\partial D(n) \geqslant 500\ \text{ms})$ OU $(NV(n) = \text{'EOF'})$
*ALORS* $NV(n) \leftarrow \text{'DP'}$

$$NV(n-1) \leftarrow \text{'FP'} + \text{'FM'}$$

*Commentaire :* Un grand allongement de la durée entre deux noyaux vocaliques indique un début de phrase à hauteur du deuxième NV et donc la fin de la phrase précédente pour le premier NV. La fin de phrase correspond en général à une fin de mot lexical. Le cas d'une seule phrase est traité par la détection de 'EOF' (End of File) dans le fichier signal.

Ph2. *SI* $(\partial D(n) \geqslant 300\ \text{ms})$ ET $(F_c(n-1) < 2)$ ET $(E_c(n-1) < 2)$
*ALORS* $NV(n) \leftarrow \text{'DP'}$

$$NV(n-1) \leftarrow \text{'FP'} + \text{'FM'}$$

*Commentaire :* Dans cette règle on tolère un allongement moins important que dans la règle précédente mais on impose deux niveaux bas, pour l'énergie et la fréquence fondamentale (ceci ne fonctionne que dans le cas des phrases énonciatives).

(*b*) Détection de mot grammatical

Ces règles sont applicables entre les deux frontières de la phrase 'DP' et 'FP' positionnées précédemment et permettent d'étiqueter certains noyaux vocaliques 'MG' (hypothèse de Mot Grammatical). Les mots grammaticaux sont souvent monosyllabiques : dans le cas contraire on ne fait pas la distinction entre début et fin de mot grammatical.

MG1. *SI* $(F_c(n) - F_c(n-1) \leqslant -2)$ ET $(\partial D(n) < 0)$ ET $(\partial D(n) - \partial D(n-1) \leqslant -50\ \text{ms})$
*ALORS* $NV(n) \leftarrow \text{'MG'}$

$$NV(n-1) \leftarrow \text{'FM'}$$

*Commentaire :* Le fondamental baisse fortement (saut de 2 niveaux au moins) et s'accompagne d'une accélération de la durée d'au moins 50 ms : il y a présomption dans ce cas d'un mot grammatical (noté MG) et donc le noyau précédent peut recevoir une fin de mot lexical (notée FM) etc. (il en est de même pour toutes les autres règles).

### 6.1.4. Résultats et discussion

La figure 11 montre, sur trois phrases, le positionnement des marques prosodiques effectué à l'aide des règles décrites ci-dessus.

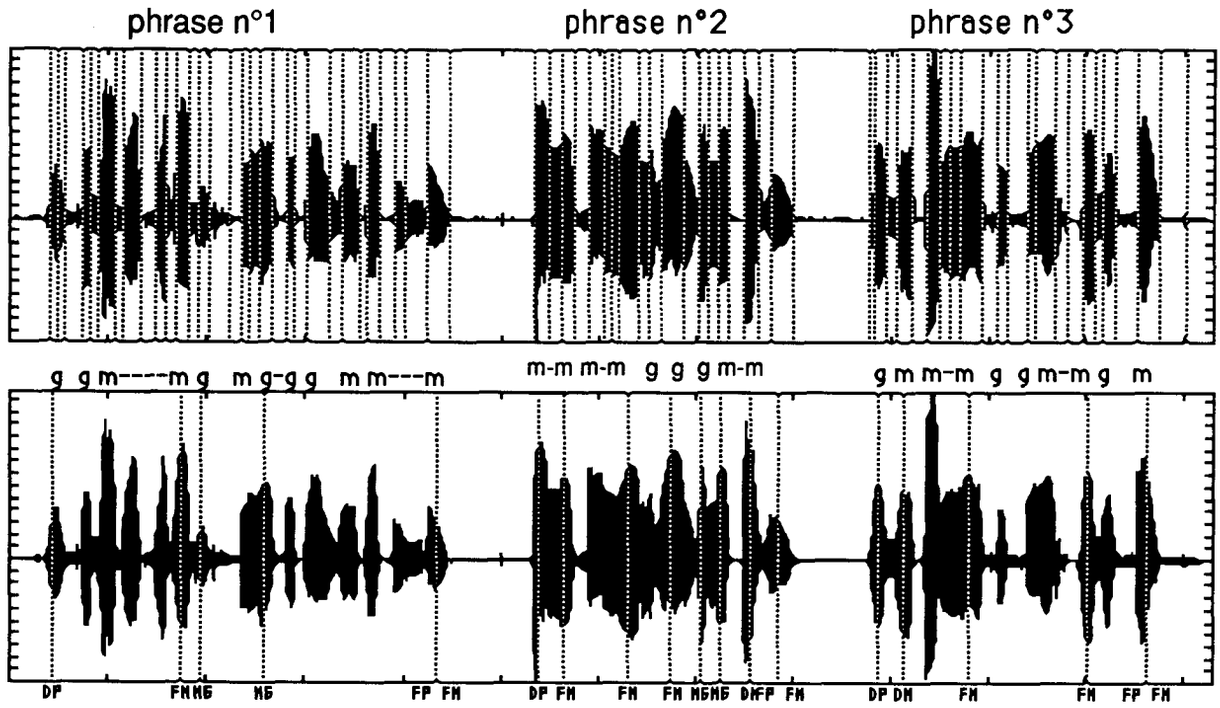



**Fig. 11. — Positionnement des marqueurs prosodiques sur les trois phrases : « il se garantira du froid avec un bon capuchon », « Annie s'ennuie loin de mes parents » et « les deux camions se sont heurtés de face » (phrases extraites du corpus PEQ de BDSON). Sont indiqués sur la figure du haut, les débuts et fins de mots grammaticaux (g) et lexicaux (m) et sur la figure du bas, les frontières détectées automatiquement DM = début de mot lexical, FM = fin de mot lexical, MG = mot grammatical, DP = début de phrase, FP = fin de phrase).**

Ces règles ont été appliquées sur 120 phrases *lues* par 3 locuteurs. Les résultats sont présentés dans le tableau 2 :

TABLEAU 2

*Résultats de détection des frontières prosodiques sur 40 phrases et 3 locuteurs*

| Etiquette | FP + FMLDP | | HMG | HDML | HFML | Total |
|---|---|---|---|---|---|---|
| Correct | 40 | 40 | 50 | 52 | 58 | 240 |
| Incorrect | 0 | 0 | 8 | 8 | 6 | 22 |
| Taux | 100 % | 100 % | 86,2 % | 86,6 % | 90,6 % | 91,6 % |

Les valeurs indiquées ci-dessus sont des taux de confiance pour les frontières détectées. Cela ne signifie donc pas que 91,6 % des frontières sont détectées, mais que parmi celles qui le sont 91,6 % sont correctes.

Ce taux de détection (ou taux de productivité des règles) est d'environ 40 % ce qui est très suffisant pour le rôle démarcatif de la prosodie assigné au système de reconnaissance. De plus, les frontières sont réparties assez uniformément le long de la phrase (*fig.* 11) ce qui assure une régularité de guidage au superviseur. Ce module prosodique est en cours d'intégration dans le système DIRA : son rôle est maintenant suffisamment clarifié pour devenir efficient dans un proche avenir. Son rôle principal sera de limiter les hypothèses lexicales en prédiction — pour les mots dont on connaît le début et la fin — et d'assurer la cohérence lexicale en vérification.

## 7. Résultats, discussion

Le système DIRA a été évalué de plusieurs manières :

(*a*) à partir de données simulées pour le DAP afin d'introduire de façon contrôlée des erreurs de plusieurs types : insertion, délétion, substitution de phonèmes en début milieu et fin de mots, pour tester la stratégie,

(*b*) à partir de corpus de sons tels que BDSONS pour les experts non linguistiques (le DAP, la prosodie) puisqu'il est impossible pour ce corpus de définir un cadre applicatif restreint,

(*c*) à partir d'une application donnée (navigation d'un robot mobile) pour les modules linguistiques (l'analyseur lexical, l'analyseur syntaxico-sémantique et la compréhension).

Les résultats pour (*b*) ont été décrits dans les chapitres dédiés aux experts concernés. Pour (*a*) et (*c*) les résultats sont commentés ci-après.

Pour la stratégie I, prise en exemple ci-dessus, les principaux problèmes restent :

— l'explosion des hypothèses lexicales développées à partir des mots monosyllabiques,
— la fragilité du développement lexical lorsque les premiers phonèmes du mot sont mal reconnus,
— le retour arrière systématique en cas d'impasse et l'élargissement des hypothèses qui augmentent in fine l'ambiguïté des messages reconnus.

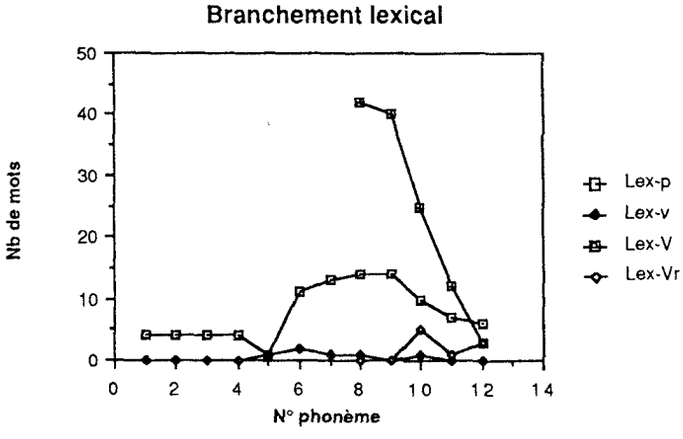
Ceci est illustré par les résultats de compréhension sur la phrase : « ouvrir la fenêtre »

TABLEAU 3

*Séquence des informations reconnues aux niveaux phonétique et lexical pour la phrase « ouvrir la fenêtre » avec les conventions suivantes :*

colonne phon : liste des phonèmes réellement prononcés,
colonne DAP-p : liste des traits reconnus pour chaque phonème. Légende : V = voyelle, dif = diffus, Z = fricative sonore, C = consonne, L = liquide, com = compact, Q = occlusive sourde, S = fricative sourde.
colonne DAP-v : idem Dap-p mais en vérification,
colonne LEX-p : nombre de mots lexicaux développés en prédiction,
colonne LEX-v : nombre de mots lexicaux acceptés en vérification,
colonne LEX-V : nombre de mots lexicaux développés après retour-arrière en vérification,
colonne EXx-Vr : nombre de mots lexicaux acceptés après retour-arrière.

| phon | DAP-p | DAP-v | LEX-p | LEX-v | LEX-V | LEX-Vr |
|---|---|---|---|---|---|---|
| u | V dif | V | 4 | 0 | | |
| v | Z | Z | 4 | 0 | | |
| R | C | C | 4 | 0 | | |
| i | V | V | 4 | 0 | | |
| R | C | C | 1 | 1 | | |
| l | C L | C | 11 | 2 | | |
| a | V com | V | 13 | 1 | | |
| f | Z | S | 14 | 1 | 42 | 0 |
| n | C | C | 14 | 0 | 40 | 0 |
| e | V dif | V | 10 | 1 | 25 | 5 |
| t | Q | Q | 7 | 0 | 12 | 1 |
| R | C | C | 6 | 0 | 3 | 3 |



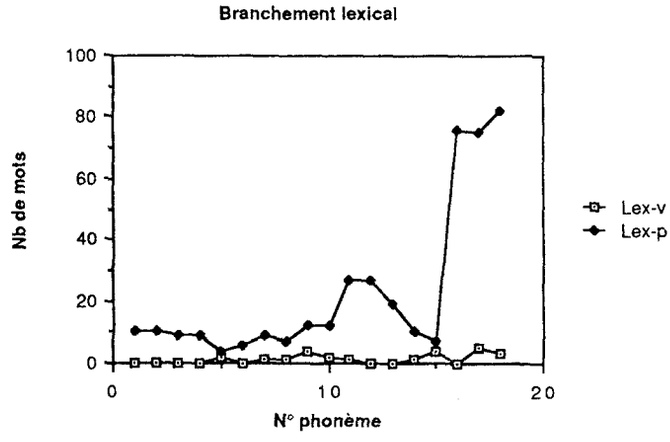

La seule phrase obtenue en fin d'analyse est : « uvRiR la fnetR » c'est-à-dire la bonne réponse (avec élision du « e » muet dans fenêtre). Un retour arrière se produit après une erreur de reconnaissance du phonème /f/ étiqueté Z = fricative sonore. Comme ce phonème est en début de mot, il déclenche une erreur dans le développement des hypothèses lexicales et oblige le superviseur, après détection d'une impasse, à un retour-arrière. On notera également le grand degré d'imprécision des traits phonétiques reconnus qui n'empêche pas la compréhension correcte de la phrase : ceci se paie par un facteur de branchement lexical assez important. Cette stratégie est donc robuste mais gère de ce fait une liste d'hypothèses importante. Elle peut être allégée chaque fois que le facteur de branchement lexical est faible c'est-à-dire en fin de mot. De même une vérification phonétique très stricte en début de mot pourrait réduire ce facteur.

On pourrait multiplier les exemples [47] :

TABLEAU 4

*Séquence des informations reconnues aux niveaux phonétique et lexical pour la phrase « avancer vers la table gauche »*

| phon | DAP-p | DAP-v | LEX-p | LEX-v | |
|---|---|---|---|---|---|
| a | V com | V | 10 | 0 | ——> verbe |
| v | Z | Z | 10 | 0 | |
| A~ | V gra | V | 9 | 0 | |
| s | S | S | 9 | 0 | |
| e | V fer | V | 4 | 2 | |
| v | Z | Z | 6 | 0 | ——> préposition |
| E | V com | V | 9 | 1 | |
| R | C L | C | 7 | 1 | |
| l | C lat | C | 12 | 4 | ——> article |
| a | V bem | V | 12 | 2 | |
| t | Q | Q | 27 | 1 | ——> nom |
| a | V ouv | V | 27 | 0 | |
| b | C O | C | 19 | 0 | |
| l | C L | C | 10 | 1 | |
| * | V gra | V | 7 | 4 | |
| g | C | C | 76 | 0 | ——> **adjectif** |
| O | V ouv | V | 75 | 5 | |
| S | S | S | 83 | 3 | |




**Les phrases reconnues sont :**

1. avA ~ se vER la tabl gOS. avancer vers la table gauche
2. avA ~ se vER la tabl* gOS. avancer vers la table gauche (deuxième variante phonologique).

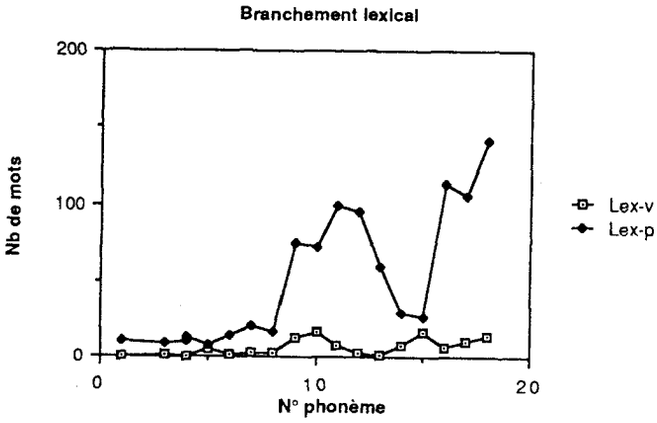
L'adjectif « gauche », mono-syllabique, provoque une certaine explosion combinatoire, réduite finalement par le faible nombre de mots acceptés.

TABLEAU 5

*Séquence des informations reconnues aux niveaux phonétique et lexical pour la phrase « avancer vers la table gauche » (deuxième variante).*

| phon | DAP-p | DAP-v | LEX-p | LEX-v | |
|---|---|---|---|---|---|
| a | V com | V | 10 | 0 | ——> verbe |
| v | Z | Z | 10 | 0 | |
| A~ | V | V | 9 | 1 | |
| s | S | S | 13 | 0 | |
| e | V | V | 8 | 5 | |
| v | Z | Z | 14 | 1 | ——> préposition |
| E | V com | V | 21 | 3 | |
| R | C | C | 17 | 3 | |
| l | C | C | 75 | 13 | ——> article |
| a | V bem | V | 73 | 16 | |
| t | Q | Q | 100 | 8 | ——> nom |
| a | V ouv | V | 95 | 2 | |
| b | C | C | 60 | 1 | |
| l | C | C | 29 | 8 | |
| * | V | V | 27 | 17 | |
| g | C | C | 114 | 6 | ——> adjectif |
| O | V ouv | V | 106 | 10 | |
| S | S | S | 142 | 4 | |

p O R t *




**Les phrases reconnues sont :**

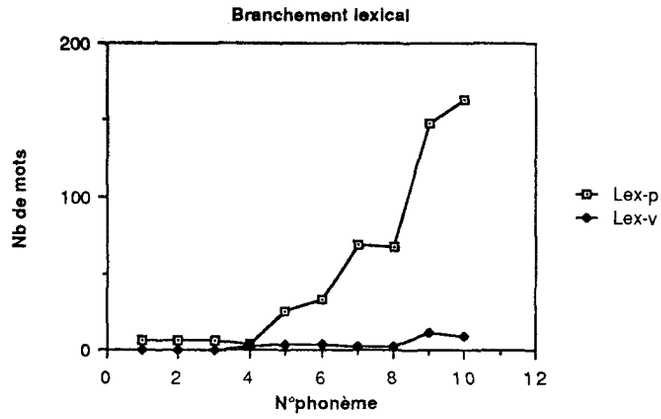
1. avA ~ se vER la tabl gOS. avancer vers la table gauche
2. avA ~ se vER la tabl* gOS. avancer vers la table gauche
3. avA ~ se vER la pORt gOS. avancer vers la porte gauche
4. avA ~ se vER la pRTt* gOS. avancer vers la porte gauche.

Dans ce cas la faiblesse des informations phonétiques se paie par une forte ambiguïté phrastique (mais le problème d'impasse est résolu).

TABLEAU 6

*Séquence des informations reconnues aux niveaux phonétique et lexical pour la phrase « longer la table ».*

| phon | DAP-p | DAP-v | LEX-p | LEX-v | |
|---|---|---|---|---|---|
| l | CL | C | 6 | 0 | ——> verbe |
| o~ | V | V | 6 | 0 | |
| Z | Z | Z | 6 | 0 | |
| e | V | V | 4 | 2 | |
| l | CL | C | 26 | 4 | ——> article |
| a | V com | V | 33 | 4 | |
| t | Q | Q | 69 | 2 | ——> nom |
| a | V com | V | 67 | 2 | |
| b | C | C | 148 | 11 | |
| l | CL | C | 163 | 9 | |




**La phrase reconnue :**

1. lo ~ Ze la tabl longer la table

## Conclusion

Ce système, par sa stratégie dynamique s'adapte à des univers d'utilisation variés. Le concept « multi-expert » le rend très modulaire et ouvert. Son superviseur, traité en planificateur de tâches, raisonne sur l'évolution de la situation au cours de la reconnaissance sans empiéter sur les décisions des spécialistes qui restent maîtres dans leur domaine de compétence. La notion de niveaux « inférieur » et « supérieur » disparaît dans cette approche et de là le problème de la dominance d'un expert sur les autres. Ces derniers n'ont aucune compétence à avoir dans leurs choix stratégiques ni dans la vision à long terme du processus de reconnaissance : ils restent dévolus entièrement à leur tâche d'expertise.

Secondairement, ce système offre une plateforme d'évaluation de stratégies de reconnaissance et de compréhension de la parole continue : il suffit pour cela de simuler le comportement de l'expert que l'on désire tester.

Le système DIRA (sans le dialogue) est opérationnel en Prolog à l'heure actuelle. Une application de robotique (500 mots, syntaxe limité) a été choisie comme terrain d'application. Moyennant la résolution du problème du temps de réponse (la puissance croissante des circuits micro-électroniques appuie notre optimisme), un tel système est envisageable à court terme pour le dialogue homme/machine.

*Manuscrit reçu le 4 octobre 1989.*

## BIBLIOGRAPHIE

[1] G. Bailly, 1986, *Détection du fondamental par traitement AMDF et programmation dynamique,* Actes 15e Journées d'Etude sur la Parole, Aix-en-Provence, pp. 213-216.

[2] T. G. Bever, *The cognitive basis for linguistic structures.* In J. R. Hayes ed., cognition and the development of language (New York), pp. 279-362.

[3] M. Bierwisch, 1985, La nature de la forme sémantique d'une langue naturelle. DRLAV, revue de ling., 33, pp. 5-24.

[4] A. Bonneau, M. Rossi, G. Mercier, 1986, *Hierarchical representation of French vowels by Expert System.* Proc. of Montreal Symposium on Speech Recognition, McGill University, pp. 20-21.

[5] J. Bresnan and R. M. Kaplan, 1982, Introduction : Grammar as mental representations of language. *The mental representations of grammatical relations* in Bresnan ed., Cambridge Mass. & London, MIT Press.

[6] R. Bulot, 1987, *Techniques de l'IA pour la reconnaissance de la parole application au décodage acoustico-phonétique.* Thèse de docteur en informatique, Université d'Aix-Marseille II, 163 p.

[7] J. Caelen, 1979, *Un modèle d'oreille. Analyse de la parole continue. Reconnaissance phonémique.* Thèse d'État Sciences, Toulouse.

[8] J. et G. Caelen, 1981, *Indices et propriétés dans le projet ARIAL II.* Séminaire « Processus d'encodage et de décodage phonétique » Toulouse, pp. 128-143.

[9] J. Caelen, 1988, *Meta-stratégie en reconnaissance dans le système DIRA-RAP.* Actes 17e Journée d'Etudes sur la Parole, Nancy, pp. 173-179.

[10] J. Caelen, H. Tattegrain, H. Meloni, R. Bulot, G. Mercier, A. Bonneau, 1988, *Une base de règles pour le décodage acoustico-phonétique : le cas des occlusives sourdes.* Séminaire de décodage acoustico-phonétique à Nancy, le 23 septembre 1988.

[11] G. Caelen-Haumont, 1981, *Structures prosodiques de la phrase énonciative simple et étendue.* Hamburger Phonetische Beitrage, Band 34, Hamburg Buske.

[12] G. Caelen-Haumont, 1986, *Grammatical components and macro-prosody : quantitative analysis toward statistical correlations.* Proceedings of the Montreal Symposium on Speech Recognition, Montreal, Canada, pp. 82-84.

[13] S. A. Cerri, P. Landini and M. Leoncini, 1987, *Cooperative agents for knowledge-based information systems.* AII, Vol. 1, n° 1, pp. 1-24.

[14] N. Chomsky, 1980, *Rules and representations.* New York : Columbia University Press.

[15] D. Clément, 1985, *Syntaxe et compétence, syntaxe et performance, syntaxe cognitive ?* DRLAV, revue de linguistique, 33, pp. 53-90.

[16] W. E. Cooper, J. Paccia-Cooper, 1980, *Syntax and speech.* Harvard University Press, Cambridge.

[17] P. Delattre, 1966, *Les dix intonations de base du français.* French Review 40(1), pp. 1-14.

[18] R. De Mori, L. Lam and M. Gilloux, 1987, *Learning and Plan Refinement in a Knowledge-Based System for Automatic Speech Recognition.* IEEE-PAMI, Vol. 9, n° 2, pp. 289-305, March.

[19] L. D. Erman, F. Hayes-Roth, V. R. Lesser and D. R. Reddy, 1980, *The Hearsay-II speech understanding system : Integrating knowledge to resolve uncertainty.* Comput. Surv., Vol. 12, pp. 213-253.

[20] J. Ferber et M. Ghallab, 1988, *Problématiques des univers multi-agents intelligents.* PRC/IA actes des journées nationales, Teknea éditeur, pp. 295-320.

[21] C. Fillmore, 1971, Types of lexical information, in *Semantics : an interdisciplinary reader.* Cambridge University Press, Steinberg and Jakobovits, pp. 370-392.

[22] R. E. Fikes and T. Kehler, 1985, *The role of frame-based representation in reasoning.* Com. ACM, 28(9), pp. 904-920.

[23] J. D. Fodor, 1983, *The Modularity of Mind. Cambridge,* MA : the MIT Press.

[24] D. Fohr, J. P. Haton, Y. Laprie, F. Lonchamp, J. M. Pierrel, 1988, *Paramétrisation acoustique et décodage phonétique fondé sur des connaissances, pour la parole continue multilocuteurs.* Actes du séminaire « Décodage acoustico-phonétique », L. Miclet éd., GRECO-PRC « Communication Homme-Machine », pp. 29-34.

[25] K. I. Forster, 1979, *Levels of processing and the structure of the language processor.* In W. E. Cooper and E. C. T. Walker (eds.), Sentence Processing : Psycholinguistic Studies, pp. 216-225.

[26] C. Garbay et S. Pesty, 1989, *MAPS : un Système Multi-agents pour la Résolution de Problèmes.* Actes du 7e congrès AFCET, RFIA, tome 1, pp. 355-368.

[27] Y. F. Gong and J. P. Haton, 1988, A Specialist Society for Continuous Speech Understanding. ICASSP, New York, April.

[28] M. Halle, 1985, *Speculations about the representation of words in memory.* In V. A. Fromkin ed., Phonetic Linguistics, Academic Press, New York, pp. 101-114.

[29] J. P. Haton, 1984, *Present Issues in Continuous Speech* Recognition and Understanding NATO Advanced Study Institute, Bonas, juillet.

[30] F. Hautin et A. Vailly, 1986, *La coopération entre systèmes experts.* Journées nationales PRC/IA, Cepadues éditions.

[31] C. Hewitt et W. A. Kornfeld, 1981. *The scientific community metaphor.* IEEE Trans. on Man, Systems and Cybernetics, Vol. CMC 11 (1).

[32] R. M. Kaplan and J. Bresnan, 1982, Lexical-Functional Grammar : a formal system for grammatical representation. *The mental representations of grammatical relations* in Bresnan ed., Cambridge Mass. & London, MIT Press.

[33] J. I. Kiger, A. L. Glass, 1983, *The facilitation of lexical decisions by a prime ocurring after the target.* Memory and Cognition, 11, 356-365.

[34] D. H. Klatt, 1977, *Review of the ARPA Speech understanding Project.* JASA Vol. 62, n° 6, December.

[35] D. H. Klatt, 1986, *The problem of variability in speech recognition and in models of speech perception.* In J. Perkell and D. Klatt eds., « Variability and Invariance in Speech Processes », Erlbaum.

[36] D. H. Klatt, 1986, *Models of phonetic recognition I : Issues that arise in attempting to specify a feature-based strategy for speech recognition.* Proc. of the Montreal Symposium on Speech Recognition, McGill University, pp. 63-66.

[37] K. Konolidge et N. J. Nilsson, 1980, *Multi-agent planning systems.* Proc. AAAI-80, Stanford, pp. 138-142.

[38] W. A. Lea, 1980, *Trends in Speech Recognition,* Prentice-Hall, 1980.

[39] W. A. Lea, M. F. Medress & T. E. Skinner, 1975, *A prosodically guided speech understanding strategy*. IEEE Trans. on Acoust. Speech and Sig. Procs., ASSP. Vol. 23, *1*, pp. 30-38.

[40] W. A. Lea, F. Clermont, 1984, *Algorithms for acoustic prosodic analysis*. Proc. ICASSP Vol. 3, pp. 42.7.1-42.7.4.

[41] J. F. Le-Ny, 1979, *La sémantique psychologique*. Press Universitaire, Paris.

[42] H. Méloni, R. Bulot, 1986, *Un système de traitement de connaissances pour le décodage acoustico-phonétique*. Proc. of the Montreal Symposium on Speech Recognition McGill University, pp. 26-27.

[43] G. Mercier, M. Gerard, M. Gilloux M et C. Tarridel, 1984, *Acoustic Phonetic decoding in the SERAC Expert System*. Actes du séminaire Franco-Suédois Grenoble, avril 1985.

[40] G. Mercier, A. Cozannet et J. Vaissière, 1988, *Recognition of speaker-dependent continuous speech with Keal-Nevezh*. In : Recent Advances in Speech Understanding and Dialog System, NATO ASI Series, Nieman, Lang & Sagerer, ed., Springer Verlag.

[45] M. Minsky, 1975, *A framework for representing knowledge*. In The Psychology of Computer Vision. P. Winston ed., New York : McGraw Hill.

[46] P. Moussel, J. M. Pierrel, A. Roussanaly, *Coopération entre syntaxe, sémantique et pragmatique dans un système de dialogue oral H/M*. AFCET, 7e congrès RFIA, Paris, pp. 371-386.

[47] M. K. Nasri, G. Caelen-Haumont, J. Caelen, 1989, *Using prosodic rules in speech recognition expert system*. Proc. IEEE-ICASSP, Glasgow.

[48] E. Reynier et J. Caelen, 1989, *ATN Compiler and Parser for an ASR system*. Proc. of EUROSPEECH'89, Paris, pp. 398-401.

[49] M. Rossi, M. Chaffcoulof, 1972, *Les niveaux intonatifs*. Travaux de l'Institut de phonétique d'Aix-en-Provence, pp. 167-176.

[50] G. Sabah, 1988, *L'intelligence artificielle et le langage*. Vol. 1, Hermès, Paris.

[51] K. N. Stevens, 1986, *Models of phonetic recognition II :* An approach to feature-based recognition. Proc. of the Montreal Symposium on Speech Recognition, McGill University, pp. 67-68.

[52] J. Vaissière, 1983, *A suprasegmental component in a French speech recognition system : reducing the number of lexical hypotheses and detecting the main boundary*. Recherches Acoustiques, CNET, Vol. VII, pp. 109-125.

[53] N. Vigouroux et J. Caelen, 1985, *Segmentations en vue de l'organisation d'une base de données acoustiques et phonétiques*. Actes 14e Journée d'Etudes de la Parole, SFA, Paris, pp. 152-155.

[54] D. E. Wilkins, 1984, *Domain-independant Planning : representation and plan generation*. Artificial Intelligence, Vol. 22, n° 3, pp. 269-302, April.

[55] T. Winograd, 1983, *Language as a cognitive process*. Vol. 1 : Syntax-Reading, Mass., etc., Addison-Wesley ed.

[56] M. Withgott, M. A. Bush, 1986, *On the robustness of phonetic information in short-time speech spectra*. Proc. of the Montreal Symposium on Speech Recognition, McGill University, pp. 101-102.

[57] J. Wolf, W. Woods, 1980, *The HWIN speech understanding system, in Trends in speech recognition*. Prentice-Hall, Englewood Cliffs, pp. 316-339.

[58] W. Woods, 1978, *Generalization of ATN grammars*. In Research in natural language understanding, Woods and Brachman, BBN report N° 3963, Cambridge, MA.

[59] W. Zue, 1986, *Models of phonetic recognition III : The role of analysis by synthesis in phonetic recognition*. Proc. of the Montreal Symposium on Speech Recognition, McGill University, pp. 69-70.