



THESE DE DOCTORAT DE 3^{ème} CYCLE

LINGUISTIQUE

DEPARTEMENT DE RECHERCHES LINGUISTIQUES

UNIVERSITE DE PARIS-VII

Daniel MEMMI

"GENERATION AUTOMATIQUE DE PHRASES"

OCT. 1979

Directeur de Recherche : M. M.GROSS

Membres du Jury :	M. M.GROSS	Président
	M. J.S.LIENARD	Examineur
	M. G.RENARD	Examineur
	M. M.SALKOFF	Examineur



5 rue Saint-Merri
75004 PARIS
278-02-63

Samuel

1046

58 v. h. les

"notions de"

comme notions?

59 les concepts de

sont des notions

ou des notions

ou des notions

ou des notions

ou des notions

p. 36 recours

59 Allen manger

1046

48 02

49 comment

50

51

52

53

54

55

56

57

58

59

60

61

62

63

THESE DE DOCTORAT DE 3ème CYCLE

LINGUISTIQUE

195-90 de la les

et les complètes

de la les et

d'elles et de la les

et de la les et de la les

DEPARTEMENT DE RECHERCHES LINGUISTIQUES

UNIVERSITE DE PARIS-VII

1-80 de la les "syntactique"

pour la grammaire orale

1979

MEM

Daniel MEMMI

"GENERATION AUTOMATIQUE DE PHRASES"

OCT. 1979

Directeur de Recherche : M. M.GROSS

Membres du Jury :	M. M.GROSS	Président
	M. J.S.LIENARD	Examineur
	M. G.RENARD	Examineur
	M. M.SALKOFF	Examineur



13155900 ND

Je tiens à remercier tous ceux qui au D.R.L. et au L.I.M.S.I. m'ont aidé à effectuer ce travail et m'ont permis de le mener à bien :

MM. Chastagner, Culioli, Mme Dequéant, MM. Gross, Liénard, Malavard, Mariani, Osorio, Renard, Salkoff, Teil.

Sans eux cette thèse n'aurait pas pu être réalisée.

SUMMARY

This doctoral thesis in computational linguistics is a contribution to both linguistic analysis and computer processing of natural language in its spoken form. After a general survey of man-machine communication by speech, is described a computer program devised to generate phonetic sentences from semantic formulas.

The program takes as input the meaning given in a logical formulation and then generates the corresponding spoken sentences in French. The output sentences are produced directly in phonetic code, prosody included, so as to drive a speech synthesizer.

In this way one can generate new messages expressing any meaning without having to record them beforehand. And this direct translation from meaning to speech is part of a global research project to achieve a spoken dialog with a computer system.

It should be noted that this program deals exclusively with the spoken language without any reference to the written form. It was thus necessary to work out an original systematic analysis of spoken French, which turns out to be quite distinct from the conservative written language that is usually described.

TABLE DES MATIERES

<u>INTRODUCTION</u>	1
1) Sujet de la thèse	
2) Linguistique et informatique	
3) Plan de la thèse	
<u>I- DIALOGUE HOMME-MACHINE</u>	8
1) Dialogue écrit	
1.1 Travaux syntaxiques	
1.2 Systèmes sémantiques	
2) Dialogue oral	
2.1 Projet ARPA	
2.2 Projets de dialogue	
3) Projet du LIMSI	
3.1 Modules d'entrée	
3.2 Modules de sortie	
<u>II- TRAITEMENT DE LA PAROLE</u>	18
1) Analyse	
1.1 Analyse spectrale	
1.2 Moyens d'analyse	
1.3 Phonétique acoustique	
2) Synthèse	
2.1 Synthétiseurs articulatoires	
2.2 Synthétiseurs acoustiques	
2.3 Synthèse à partir de texte	
3) Reconnaissance	
3.1 Identification phonétique	
3.2 Niveaux supérieurs	

<u>III- GENERATION DE PHRASES</u>	48
---	----

- 1) Introduction
 - 1.1 Génération automatique
 - 1.2 Programme de génération
- 2) Structures de listes
 - 2.1 Organisation des données
 - 2.2 Opérations élémentaires
- 3) Représentation sémantique
 - 3.1 Choix d'une représentation
 - 3.2 Notation fonctionnelle
 - 3.3 Arborescence sémantique
- 4) Dictionnaire
 - 4.1 Entrées lexicales
 - 4.2 Structure du dictionnaire

<u>IV- GRAMMAIRE ORALE</u>	66
----------------------------------	----

- 1) Morphologie
 - 1.1 Verbes
 - 1.2 Adjectifs
 - 1.3 Autres mots
- 2) Syntaxe
 - 2.1 Accords
 - 2.2 Placement
 - 2.3 Marqueurs prosodiques
- 3) Liaisons
 - 3.1 Liaisons consonantiques
 - 3.2 Elisions et contractions

<u>CONCLUSION</u>	87
-------------------------	----

- 1) Résultats
- 2) Limitations
- 3) Perspectives

<u>BIBLIOGRAPHIE</u>	91
----------------------------	----

INTRODUCTION

1) SUJET DE LA THESE

Le travail présenté dans cette thèse se situe parmi les recherches sur le traitement automatique du langage naturel. Il a consisté à concevoir et à réaliser un programme de génération automatique par ordinateur de messages parlés à partir de formules sémantiques. Ce programme permet donc de passer du sens d'un message à sa prononciation en langue naturelle. Les énoncés sémantiques sont donnés dans une formulation de type logique, et le programme génère les phrases correspondantes en français parlé, c'est-à-dire en notation phonétique pour commander un synthétiseur de parole.

On veut ainsi rendre la machine capable d'émettre des messages exprimant un sens donné sans devoir les enregistrer au préalable. La variété des messages possibles devient alors très grande, sinon même illimitée grâce à la richesse combinatoire du langage naturel. Il est donc intéressant, tant sur le plan pratique, que d'un point de vue linguistique, d'essayer de simuler la production du langage par un locuteur humain. D'ailleurs ce programme est partie intégrante d'un projet global de dialogue homme-machine par la parole élaboré au L.I.M.S.I. à Orsay.

Il se place donc aussi dans le cadre des travaux sur l'analyse, la synthèse et la reconnaissance automatique de la parole. Ce domaine de recherche connaît actuellement un développement important, qui permet d'escompter pour un avenir proche de pouvoir dialoguer par la parole avec un ordinateur, au moins dans un domaine donné. Mais la parole pose des problèmes particuliers encore plus difficiles que ceux rencontrés dans le traitement de texte écrit. Or les descriptions grammaticales se limitent le plus souvent à la langue écrite, qui diffère sensiblement de la langue parlée. L'approche orale de ce travail a donc demandé une description linguistique originale.

.../...

Plus généralement on peut rattacher le traitement automatique du langage au domaine connu sous le nom d'intelligence artificielle. Certes, cette expression n'est pas très bonne, car elle est à la fois vague et pompeuse et recouvre un ensemble de recherches très diverses comme la documentation automatique, la reconnaissance des formes, la programmation des jeux, la démonstration automatique de théorèmes, et la résolution de problèmes en général... Mais toutes ces recherches ont en commun le fait qu'elles visent à l'automatisation de processus intellectuels. Ce sont des tentatives pour étendre le champ d'action des ordinateurs, en les rendant capables d'effectuer des traitements de plus en plus complexes qui exigeaient jusqu'ici une intelligence humaine.

En ce sens, il s'agit là d'une extension toute naturelle de l'informatique, qui a débuté par la formalisation et l'automatisation de procédés élémentaires de calcul et de raisonnement. Mais précisément cette extension à de nouveaux domaines demande une analyse approfondie des processus que l'on cherche à reproduire en machine, d'où son intérêt théorique autant que pratique.

2) LINGUISTIQUE ET INFORMATIQUE

Ce travail se situe donc au carrefour de la linguistique et de l'informatique. On peut considérer aussi bien qu'on a utilisé des moyens informatiques pour faire des recherches linguistiques, ou qu'on a appliqué la linguistique à des problèmes d'informatique. Mais quelle que soit la motivation de départ, on obtient un outil de recherche remarquable en utilisant l'informatique dans un domaine mal formalisé. En effet, les sciences humaines donnent encore trop souvent lieu à des spéculations d'ordre littéraire ou philosophique plutôt qu'à des recherches rigoureuses. L'informatique peut alors apporter un moyen de formaliser des processus dynamiques complexes dans un langage à la fois précis et puissant.

Car pour simuler un phénomène sur ordinateur, c'est-à-dire pour écrire un programme qui le reproduise, il faut en explorer tous les détails jusqu'à ce qu'on obtienne un résultat satisfaisant. On a là un critère de vérification de la validité d'une analyse, et on s'aperçoit alors très vite que les descriptions verbales habituelles sont tout à fait insuffisantes, sinon fausses ou inutiles. On est obligé de dégager bien des faits nouveaux avant que le programme puisse fonctionner correctement.

On se rapproche ainsi des exigences de rigueur, de précision et d'exhaustivité qui règnent dans les sciences exactes et leur donnent leur efficacité.

Or la linguistique de ces dernières années pourrait profiter d'une approche plus pratique et plus rigoureuse. Après les succès prometteurs de la linguistique structurale, tant dans l'école de Prague que chez les structuralistes américains, il semble que la situation se soit dégradée du point de vue méthodologique. Alors que les travaux structuralistes en phonologie nous paraissent dans l'ensemble être restés satisfaisants, la grammaire générative et transformationnelle de l'école de Chomsky, qui est devenue dominante depuis lors, a mené à des spéculations peu convaincantes. De fait la situation actuelle paraît plutôt confuse et divisée.

Certes les résultats de la phonologie structurale sont sans doute dûs autant au domaine (il n'y a jamais qu'un petit nombre fini de phonèmes dans une langue) qu'à la méthode elle-même, et les recherches étaient limitées en syntaxe et presque inexistantes en sémantique. Chomsky a eu le mérite indiscutable d'aborder de front les problèmes de la syntaxe et de la sémantique et d'y sensibiliser la communauté des linguistes, tout en apportant un point de vue novateur. La grammaire générative restera une tentative intéressante de formalisation de la syntaxe, et a stimulé les réflexions sur les rapports entre la syntaxe et la sémantique. Les travaux effectués dans ce cadre ont certainement montré des faits nouveaux et enrichi la discussion linguistique dans des domaines complexes.

Malheureusement l'école générativiste a de plus en plus préféré les discussions théoriques abstraites à la recherche empirique. Et la théorie a pris une allure philosophique idéaliste : Chomsky se réfère explicitement au rationalisme cartésien pour défendre la thèse de l'innéité d'universaux linguistiques. C'est une hypothèse intéressante et défendable, mais sans rapport direct avec les problèmes proprement linguistiques dans l'état actuel des connaissances. On aboutit alors à des discussions sans aucun contenu empirique entre des positions également invérifiables car trop abstraites. Les arguments précis quand il y en a se basent sur des observations éparses et non exhaustives, sans aucun souci d'un inventaire préalable des faits pertinents.

Il devient ainsi impossible de décider de façon non arbitraire entre des propositions purement théoriques, tandis que le travail indispensable d'accumulation des données est négligé. Bien entendu et comme trop souvent dans les sciences humaines, le dogmatisme croît avec l'impossibilité de justifier réellement ces spéculations invérifiables. En fait ce système est basé moins sur des faits que sur le désir de soutenir des prises de positions philosophiques innéistes. Ce mélange inextricable de philosophie avec l'analyse proprement linguistique est absolument contraire à l'attitude scientifique, qui demanderait au contraire l'objectivité devant les faits et la suspension du jugement sur les questions invérifiables.

Sans nier l'utilité des travaux générativistes pour l'évolution de la linguistique, il est temps de revenir à une attitude plus empirique. Il serait bon de faire moins de métaphysique et de consacrer plus de temps à l'analyse linguistique proprement dite. Car si la théorie est nécessaire pour systématiser les faits observés et amener à en découvrir d'autres, elle n'a aucune signification en l'absence de données suffisantes et n'est plus qu'une construction spéculative qui ne correspond à rien de réel. Or en principe une théorie cherche à représenter quelque chose d'autre qu'elle-même...

Bien sûr on peut chercher à développer des modèles formels en soi : c'est ce que font les mathématiques et la logique, et il serait absurde d'en nier l'importance. Mais justement le mathématicien ne prétend pas représenter le réel, même s'il est en fait souvent motivé par des problèmes concrets. Il se consacre essentiellement à l'élaboration de modèles formels, et il laisse au physicien le soin d'utiliser ces modèles pour représenter les faits expérimentaux (même si la théorie suggère souvent des expériences). L'histoire des sciences montre que théorie et expérimentation ont toujours progressé de pair, en se nourrissant réciproquement. Si la théorie est indispensable pour représenter les faits, elle n'a pas de sens sans les faits.

Pour sortir du formalisme stérilisant de la grammaire générative, il est nécessaire de collecter systématiquement des données empiriques avant de chercher à en induire une formulation théorique. Cela est d'autant plus souhaitable que les exceptions semblent bien tout aussi caractéristiques du langage naturel que les règles. Il est donc impossible de déterminer la généralité d'une règle quelconque sans faire un inventaire

systematique de ses conditions d'application. C'est une telle approche empirique que défend par exemple M. Gross dans "Méthodes en Syntaxe", où on trouvera un exposé plus approfondi du problème. On a donc commencé à percevoir une réaction dans ce sens.

Une autre voie de recherche possible passe justement par le traitement automatique du langage qui exige aussi une analyse concrète, rigoureuse et exhaustive. C'est du moins ce qu'on a essayé de pratiquer dans ce travail. Car cette approche technique comporte certainement un danger, ou plutôt une ambiguïté fondamentale : cherche-t-on l'efficacité avant tout, ou bien une simulation aussi fidèle que possible des processus naturels ? Les deux attitudes sont possibles, mais on ne peut les juger avec les mêmes critères (voir Pitrat à ce sujet). Quand on cherche autant à faire fonctionner un système qu'à construire un modèle, on risque de tendre vers un pragmatisme à courte vue et de faire passer une solution techniquement efficace pour une analyse plus profonde du phénomène étudié. Il suffit de rappeler qu'un avion ne bat pas des ailes pour voler !

Malgré tout, quand il s'agit de simuler des processus intellectuels complexes plutôt qu'une activité matérielle, il est fort peu vraisemblable de réaliser un programme correct (demandant un temps de calcul et un espace mémoire raisonnables) sans une analyse précise et fidèle du processus à reproduire. Et même si on décide de privilégier l'efficacité technique, on aura beaucoup appris au cours de la mise au point du modèle. Il serait certes naïf de croire qu'on peut se passer de théories, mais la construction d'un modèle informatique à partir de données suffisantes apporte déjà une vue claire des questions traitées. Ceci est particulièrement précieux dans le domaine intellectuel où l'expérimentation directe paraît encore impossible actuellement.

Ainsi cette approche pratique devrait renouveler la réflexion. Le contact constant avec des problèmes concrets peut paraître ingrat, mais apporte un autre point de vue. Car il faut résoudre des problèmes nouveaux ou simplement négligés jusqu'ici, et bien des considérations théoriques apparaissent alors inadéquates. De plus on s'aperçoit que certaines questions sont tout simplement dénuées de sens ou insolubles parce que mal posées. Ce changement d'orientation permet à la fois d'aborder des questions nouvelles et de sortir de discussions sans issue, et il devrait donc être fructueux.

.../...

Par exemple les moyens électroniques d'analyse et de synthèse ont renouvelé la phonétique en permettant l'étude acoustique de la parole. Puis les recherches sur la reconnaissance de la parole ont apporté une réflexion précise sur la perception et la compréhension du langage, qui n'aurait pas eu lieu autrement. Les besoins pratiques de la simulation ont mené à l'élaboration de modèles psycho-linguistiques détaillés de la communication parlée. Et l'attention portée à l'aspect oral du langage oblige à des analyses systématiques de la langue parlée, alors qu'on ne décrit généralement que la langue écrite, d'usage pourtant moins immédiat. Ainsi on enrichit les connaissances sur le langage.

Ensuite il est intéressant de remarquer que si on arrive à rendre les ordinateurs capables d'exécuter des tâches intellectuelles aussi complexes que le maniement du langage humain ou la démonstration de théorèmes, il ne serait plus nécessaire de séparer les activités mentales des autres processus biologiques. Car on aurait ainsi montré qu'elles sont exécutables par une machine, c'est-à-dire par un procédé purement matériel. D'ailleurs, les recherches en neurophysiologie tendent aussi de plus en plus à ramener les processus mentaux au fonctionnement matériel du cerveau, de la même façon qu'on a pu réduire les phénomènes biologiques aux lois physico-chimiques. On irait alors vers l'unification de principe des lois naturelles.

Enfin il est probable que l'automatisation de nouvelles tâches intellectuelles jusqu'ici réservées à l'homme va entraîner une extension encore plus grande de l'informatique. Or dans une société où les activités bureaucratiques sont de plus en plus importantes, le traitement automatique de l'information risque fort d'avoir des conséquences sociales et économiques étendues. Il faut espérer qu'on saura prévoir et maîtriser cette évolution technologique inévitable.

3) PLAN DE LA THESE

Mais après cette introduction théorique, on va justement passer à des réalisations concrètes en décrivant le travail réalisé. Ce travail faisant partie d'un projet global de dialogue homme-machine par la parole, on commencera dans une première partie ("Dialogue homme-machine") par donner un aperçu général de ce projet dans le cadre de la communication en langage naturel avec l'ordinateur. Puis dans une deuxième partie

.../...

.../...

("Traitement automatique de la parole") on exposera en détail les techniques particulières utilisées pour l'analyse, la synthèse et la reconnaissance automatique de la parole.

Après ces chapîtres généraux, on passera au sujet proprement dit de cette étude, la génération automatique de messages. Dans une troisième partie ("Génération de phrases"), on exposera la structure du programme réalisé, tout en discutant les solutions adoptées. Puis dans une quatrième partie ("Grammaire orale") on décrira en détail la grammaire parlée qu'on a mis au point pour le programme. Enfin dans la conclusion, on essaiera de faire le bilan des résultats et des déficiences de cette réalisation pour en tirer des indications utiles, d'autant plus qu'on a toujours essayé de voir dans leur généralité les problèmes rencontrés.

I - DIALOGUE HOMME-MACHINE

Depuis la révolution industrielle, les machines ont pris une place de plus en plus grande dans la vie quotidienne et deviennent de plus en plus complexes. Et l'avènement des machines à traiter l'information que sont les ordinateurs a encore étendu le domaine d'action du machinisme. Il devient donc de plus en plus important de pouvoir communiquer avec ces machines le plus facilement possible. On utilise pour cela une variété de moyens, allant des gestes stéréotypés du conducteur automobile aux langages de programmation des ordinateurs. Mais boutons, manettes et cadrans ainsi que les langages artificiels de l'informatique exigent tous un apprentissage plus ou moins long et difficile, qui limite souvent à des spécialistes l'usage des machines.

Or nous connaissons déjà un moyen de communication commode et puissant : le langage naturel. L'extraordinaire richesse des messages qu'il peut transmettre, et surtout le fait qu'il est familier à tous depuis l'enfance, en font un outil privilégié. De plus sous sa forme orale, le langage est d'un usage immédiat, presque instinctif, et laisse libre les mains et les yeux. Bien sûr, ce n'est pas toujours le meilleur moyen pour diriger des processus rapides et continus comme la conduite d'un véhicule. Mais dans les applications où le langage naturel est lui-même objet du traitement (la documentation automatique par exemple), il paraît difficilement remplaçable comme moyen.

Il est donc très intéressant de tenter de donner aux ordinateurs la capacité de traiter le langage. On pourra alors dialoguer avec la machine, c'est-à-dire lui fournir de l'information et obtenir des réponses en langue naturelle. Cela rendrait l'informatique actuelle plus accessible à l'utilisateur et ouvrirait sans doute de nouvelles applications pour un public non spécialisé de plus en plus large. Et cette banalisation de la communication permettrait en plus d'exploiter directement l'énorme quantité d'information contenue dans les documents écrits.

.....

Mais évidemment le traitement automatique du langage naturel exige des connaissances détaillées sur le langage, et pose nombre de problèmes tant théoriques que pratiques. Or ces problèmes linguistiques et informatiques sont souvent loin d'être résolus et demanderont des recherches fondamentales approfondies. De fait il faut admettre que les résultats sont encore modestes malgré l'importance des moyens mis en oeuvre. Pourtant le domaine progresse et la problématique a un effet d'entraînement sur la recherche fondamentale.

Ensuite il faut distinguer le langage écrit du langage oral. La parole demande des traitements complexes pour en reconnaître les composantes, alors que les éléments sont donnés en clair dans le texte écrit, car la forme sonore est bien différente de la forme graphique. De plus langue écrite et langue orale constituent des codes apparentés mais distincts, bien qu'on ne décrive généralement que la langue écrite. Enfin les conditions d'utilisation ne sont pas les mêmes, car la parole s'emploie normalement dans un contexte de dialogue immédiat, qui amène par exemple à utiliser des phrases mal formées ou inachevées qu'on évitera dans la langue écrite.

En conséquence le traitement du langage parlé est plus limité, car on bute sur le problème difficile de la reconnaissance de la parole. Mais langue écrite et langue orale ont bien sûr beaucoup en commun, et on peut raisonnablement espérer utiliser bien des acquis des travaux sur l'écrit pour le traitement du dialogue oral. Donc après avoir décrit les recherches sur le traitement du dialogue par écrit, on exposera les problèmes du dialogue oral. Puis on détaillera le projet de dialogue oral du LIMSI dont fait partie notre programme de génération de phrases.

Mais dans tous les cas pour pouvoir communiquer avec un ordinateur, il faut qu'il puisse analyser les informations qu'on lui fournit, puis résoudre le problème posé, et enfin donner une réponse utilisable. C'est ce qui se passe quand on se sert d'un langage de programmation, et ce schéma reste vrai si on veut communiquer en langage naturel. Mais l'analyse morpho-syntaxique et sémantique devra être beaucoup plus complexe, la résolution de problèmes beaucoup plus générale, et on devra donner la réponse en langage naturel. Tout ceci est bien sûr difficile et en pratique on n'y arrive que dans des contextes très restreints, car le langage naturel est bien différent des langages formels....

.../...

1) DIALOGUE ECRIT

Le moyen actuellement le plus utilisé pour communiquer avec les ordinateurs est le texte écrit : l'entrée se fait par un clavier et la sortie sur imprimante, télétype ou écran cathodique. Mais les commandes sont données dans des langages artificiels divers, plus ou moins difficiles à manier et de puissance limitée. Lorsqu'on a voulu utiliser le langage naturel, on a naturellement commencé par traiter la forme écrite qui peut passer par le même matériel déjà standard, ce qui évite les problèmes de reconnaissance inévitables pour la parole. Cependant on peut aussi maintenant reconnaître assez bien les caractères imprimés, et donc lire automatiquement un document ordinaire, alors que la reconnaissance d'écriture manuscrite reste un problème de recherche.

Il y a déjà un grand nombre de réalisations de traitement du langage écrit, dans des domaines variés (traduction automatique, interrogation de fichiers, indexation automatique,...) et avec des résultats divers. Mais on a rarement abouti à des produits opérationnels, sauf pour des applications particulières. On peut distinguer plus ou moins arbitrairement entre les travaux purement syntaxiques et ceux qui essayent de tenir compte de la sémantique du texte.

1-1 Travaux syntaxiques

Au début de la traduction automatique on n'a d'abord fait que du mot à mot avec un dictionnaire bilingue, et en documentation automatique on s'est souvent contenté de résumer un texte par des mots-clés. Mais on s'est vite aperçu que la syntaxe était indispensable, car les mots ne prennent un sens précis qu'en contexte, et l'ordre des mots est évidemment important. On peut alors avec des analyses syntaxiques plutôt simples obtenir des résultats assez impressionnants, mais toujours dans des domaines très limités.

Du point de vue de l'outillage utilisable, il faudrait citer les travaux sur les grammaires formelles et les nombreux algorithmes d'analyse syntaxique (par exemple les réseaux de transition de Woods). Mais ces divers outils sont utilisés avec le plus grand pragmatisme, et des approches simplistes peuvent se révéler tout à fait efficaces si on s'est fixé un but limité. Il faut seulement bien voir que les résultats ne seront

.../...

pas toujours généralisables à d'autres problèmes.

En fait des programmes comme "Eliza" de Weizenbaum (avec lequel on peut avoir par écrit un entretien psychologique) se contentent pour répondre de ré-arranger l'ordre syntaxique de la phrase d'entrée, sans aucune compréhension réelle du texte. L'impression de vraisemblance du dialogue est troublante, mais assez factice, puisque le programme ne peut que ressortir ce qu'on y a entré. Et on peut en dire autant de la plupart des travaux de documentation automatique.

Pour vraiment répondre à des questions de façon intelligente (c'est-à-dire répondre autre chose que le contenu de la question ou l'information brute gardée en mémoire) il faut des programmes capables de comprendre même partiellement la signification d'un texte, et de raisonner un minimum pour en déduire ou en inférer des réponses non prévues à l'avance. Ceci demande de traiter non seulement le lexique et la syntaxe, mais aussi la sémantique et la pragmatique des textes envisagés. C'est bien ce qu'essayent de faire des travaux plus récents.

1-2 Systèmes sémantiques

On peut discuter des définitions respectives de la syntaxe, de la sémantique, et de la pragmatique, mais en tous cas il apparaît de plus en plus que le traitement de texte demande qu'on s'attaque au sens. Il faut alors représenter le sens par une notation formelle différente du langage naturel, puisqu'on ne peut se contenter de la syntaxe de surface pour fournir des réponses non triviales. Et c'est sur cette représentation du sens qu'on travaillera pour obtenir un résultat.

Mais le problème de la représentation sémantique est loin d'être résolu. Les représentations varient beaucoup selon les chercheurs et le choix paraît bien arbitraire à l'heure actuelle. Il est possible que ces différentes représentations ne soient en fait que de simples variantes notationnelles qui se révéleront équivalentes à l'usage, mais il se pourrait aussi que cela masque des problèmes fondamentaux importants. Il faudrait pouvoir justifier les éléments sémantiques qu'on utilise....

Quoiqu'il en soit, la démarche généralement suivie consiste à analyser le texte pour le traduire dans la notation sémantique choisie, puis à faire les inférences nécessaires pour résoudre le problème posé (réponse à une question, commande d'un processus, ...).

.../...

Enfin on génère la réponse en langage naturel. On peut donc distinguer trois stades du dialogue : compréhension, raisonnement et génération. Mais ces étapes sont imbriquées dans la pratique, car on ne peut par exemple comprendre vraiment sans raisonnement.

La notation formelle utilisée est souvent une variante ou une autre du calcul des prédicats, ce qui permet de bénéficier du travail important déjà accompli par des logiciens. Résoudre un problème revient alors à démontrer un théorème grâce à des procédures comme l'algorithme de résolution de Robinson. L'exemple le plus remarquable en est sans doute le programme de Winograd qui peut dialoguer sur un univers de cubes et de pyramides de façon très convaincante. Mais encore une fois, cela n'est possible que dans un univers simple et très réduit.

D'autre part il n'est pas facile de passer du langage naturel aux formules logiques, et la déduction logique diffère du raisonnement ordinaire. On peut alors tenter d'utiliser des systèmes logiques multi-valués ou probabilistes, comme par exemple le projet "Mycin" de Stanford, pour pouvoir faire des raisonnements flous plus souples que la déduction classique. On essaye ainsi d'adapter la logique à la sémantique naturelle.

On peut aussi représenter le sens d'une phrase par un petit nombre de primitives plus élémentaires que les mots ordinaires, de manière à lever les ambiguïtés de surface. C'est ce que font Schank et son équipe, entre autres, mais le choix des primitives est difficile à justifier. Et on pourrait citer d'autres propositions, comme la notion de "frame" ou de "script", espèce de scénario indiquant ce qui est pragmatiquement prévisible dans une situation donnée. On espère de la sorte se rapprocher de la complexité du raisonnement humain, mais toujours en se limitant à un domaine restreint.

Cependant, il ne faut pas croire que la prise en compte de la sémantique et la pragmatique permette de se passer d'analyse morphologique et syntaxique, et de fait les systèmes sémantiques tiennent généralement compte de la syntaxe. On doit aussi insister sur la diversité des représentations sémantiques utilisées alors que les domaines traités sont très réduits. Et la génération se limite souvent à des formules stéréotypées. En résumé, il s'agit de problèmes fondamentaux qui sont encore très discutés et les réalisations sont intéressantes mais très restreintes.

.../...

.../...

2) DIALOGUE ORAL

Pour dialoguer par la parole avec la machine, il faut disposer d'entrées et de sorties vocales spécialisées, alors que le matériel standard suffit pour le texte écrit. Ainsi, en sortie on utilisera un synthétiseur de parole qui permet de prononcer n'importe quel message sans devoir l'enregistrer à l'avance. En entrée il faudra analyser la parole pour essayer d'y reconnaître le contenu linguistique. La parole pose donc des problèmes techniques complexes, et du moins pour la reconnaissance de parole, encore imparfaitement résolus dans la pratique.

Les chercheurs sur la parole se sont donc naturellement préoccupés davantage de ces problèmes acoustiques que des problèmes linguistiques et pragmatiques du dialogue. Comme on devait d'abord effectuer la synthèse de parole et la reconnaissance préalablement à toute autre considération, le traitement du langage parlé est dans l'ensemble moins avancé que celui du langage écrit, dans la mesure où on peut comparer ces deux formes différentes de la langue naturelle.

Mais on pourra probablement s'inspirer des connaissances acquises dans le traitement de texte écrit, en particulier en syntaxe et en sémantique. Quant à la résolution de problèmes, elle paraît indépendante de la nature écrite ou orale du dialogue avec l'homme, si on suppose résolues les difficultés d'acquisition. Il faut seulement voir que la langue parlée est assez différente de la langue écrite pour qu'on s'attende à devoir faire des adaptations à tous les niveaux, de la morphologie à la conduite du discours, si on veut essayer d'extrapoler de l'écrit à l'oral.

2-1 Projet ARPA

Alors que la synthèse de parole est maintenant opérationnelle depuis un certain temps (même s'il y a encore des progrès à faire), la reconnaissance automatique reste du domaine de la recherche, du moins dans l'ensemble. Mais le projet ARPA aux Etats-Unis a permis ces dernières années de progresser et d'avoir une vue plus claire des problèmes qui restent à résoudre.

Ce projet a fourni dans les années 70 des moyens financiers importants à une dizaine d'équipes, de manière à pouvoir réaliser des systèmes de grande taille. Parmi ces équipes on peut citer Bolt, Beranek & Newman, Carnegie Mellon University, System Development

.../...

Corporation, etc... Les résultats atteints étaient encourageants et ont montré que la reconnaissance de la parole était possible si on y consacrait des moyens suffisants.

Le projet ARPA a eu également l'intérêt d'expérimenter systématiquement l'utilisation de contraintes linguistiques et pragmatiques pour améliorer la reconnaissance acoustique. Il a donc fallu maîtriser l'organisation de ces différents niveaux de traitement, ce qui représente un problème informatique difficile. Mais on ne cherchait pas à donner de réponse vocale, et il n'y avait donc pas encore de dialogue oral.

2-2 Projets de dialogue

Cependant les progrès de la reconnaissance ont permis d'envisager l'association de la synthèse à la reconnaissance en vue de pouvoir vraiment dialoguer par la parole avec l'ordinateur. Les réalisations ne sont encore pas très avancées, mais le départ est bien pris. En intégrant les travaux effectués jusques là plus ou moins séparément sur l'entrée et la réponse vocale, et en y ajoutant des possibilités de raisonnement, on peut obtenir un dialogue entièrement oral avec la machine sur des sujets bien délimités.

Par exemple en France, il faut citer les recherches en cours au CNET et au LIMSI. Dans ces deux laboratoires on a déjà réalisés des systèmes capables de tenir un dialogue simple : ainsi l'ordinateur peut répondre à des demandes de renseignements téléphoniques, passer le correspondant demandé, fournir des informations à partir d'une base de données. Bien sûr les dialogues sont encore très rudimentaires et le domaine de la discussion toujours très limité, mais on a démontré que le dialogue oral était possible. Et en améliorant les capacités de raisonnement des systèmes, on pourra obtenir des dialogues plus complexes.

Ces recherches progressent, mais elles ont déjà apporté des résultats intéressants, particuliers au dialogue oral. Ainsi le dialogue parlé demande une reconnaissance continue de bonne qualité, et c'est un problème difficile. Mais le contexte du dialogue permet d'utiliser au maximum des contraintes linguistiques et pragmatiques pour prédire ce qui pourra être dit et faciliter ainsi la reconnaissance. Le système peut orienter le dialogue en grande partie en demandant des précisions ou la confirmation de ce qui a été prononcé, pour une meilleure compréhension. Cet aspect interactif du dialogue paraît essentiel, et se révélera sans doute de plus en plus important au cours des recherches.

.../...

3) PROJET DU LIMSI

C'est le projet de dialogue oral dans le cadre duquel se place notre programme de génération, et c'est en fonction de ce projet que le programme a été conçu et réalisé. A l'origine le groupe de recherches sur la parole du LIMSI, comprenant notamment J.S. Lienard et D. Teil, puis J. Mariani, se consacrait surtout à des travaux sur l'analyse, la synthèse et la reconnaissance automatique de la parole. Puis est venue l'idée d'intégrer ces divers travaux en un projet global de dialogue et de poursuivre dorénavant les recherches dans le sens de ce projet, en essayant de compléter systématiquement les modules nécessaires.

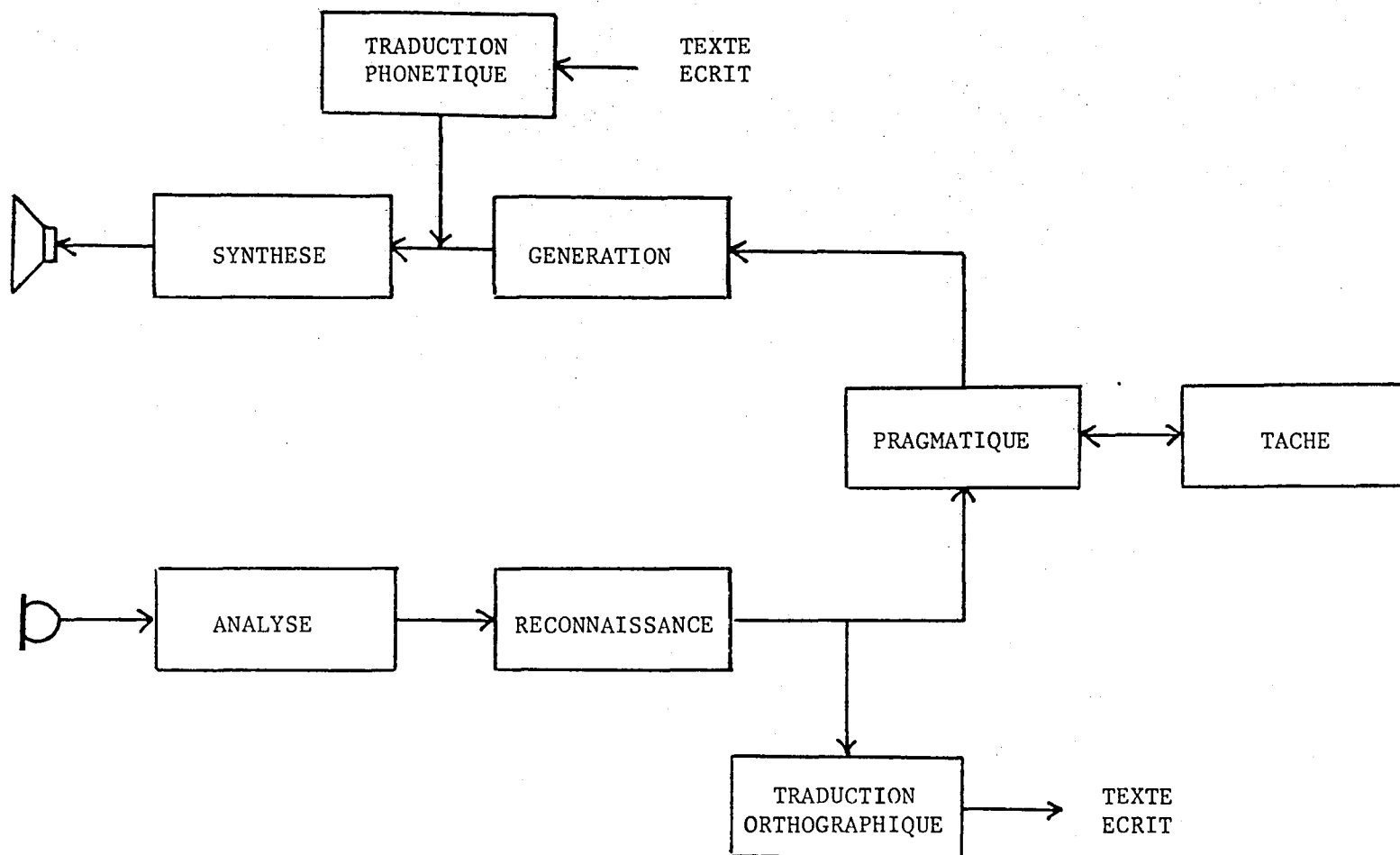
Ceci explique que nombre de modules du projet sont déjà opérationnels, surtout en synthèse, analyse et reconnaissance, puisqu'on a pu reprendre (en les adaptant) les programmes réalisés lors de recherches antérieures. C'est aussi pourquoi l'ensemble est incomplet et probablement moins cohérent que si on était parti de zéro. En particulier les modules de sortie et d'entrée sont nettement plus avancés que les niveaux supérieurs qu'il faut maintenant aborder. Et il y a souvent plusieurs versions d'un même module, correspondant à des stades divers de la recherche.

Il s'agit donc plus d'un cadre directeur assez souple que d'un projet très détaillé. Mais l'intégration des différents programmes progresse et le travail se poursuit vers les niveaux supérieurs de la syntaxe, de la sémantique et de la pragmatique. Cependant comme les possibilités de raisonnement sont encore très limitées, on peut exposer ce qui a été déjà réalisé en distinguant simplement entre modules d'entrée et modules de sortie.

3-1 Modules d'entrée

Pour pouvoir reconnaître ce qui est dit, il faut d'abord bien sûr capter la parole avec un microphone, puis faire une analyse du signal recueilli, avant toute tentative d'identification. On effectue donc une analyse spectrale grâce à des techniques diverses : bancs de filtres électroniques ou calculs numériques. On obtient ainsi un petit nombre de paramètres simples variant au cours du temps. Il faudra alors en extraire le contenu linguistique.

.../...



Projet du LIMSI

Mais le processus de reconnaissance est particulièrement complexe puisqu'on essaye d'améliorer la reconnaissance acoustique tout en tenant compte des contraintes lexicales, syntaxiques, sémantiques et pragmatiques. On peut aussi faire de la reconnaissance par mots, ce qui est plus facile si le nombre de mots est limité. Enfin on essaye d'orthographier correctement la chaîne phonétique reconnue, ce qui demande d'utiliser un dictionnaire morphologique et une syntaxe. On pourrait ainsi non seulement reconnaître la parole mais aussi la transformer en texte écrit, parallèlement au dialogue oral.

3-2 Modules de sortie

Pour que la machine puisse émettre des messages parlés, il faut générer ces messages puis les faire prononcer par un synthétiseur de parole. Cette synthèse acoustique peut se faire de plusieurs manières à l'aide des divers types de synthétiseurs électroniques du laboratoire (la série des "Icophone") ou bien par calcul numérique. Mais en tous cas il faut commander la synthèse par ordinateur pour assurer l'enchaînement des formes acoustiques. D'autre part on peut aussi partir de texte écrit grâce à des programmes de traduction phonétique convertissant l'orthographe normale en code phonétique.

Quant à la génération automatique de phrases phonétiques à partir du sens, c'est justement l'objet de ce travail. Celui-ci constitue donc une tentative pour étendre vers la sémantique la branche de réponse du système de dialogue. On exposera évidemment en détail la réalisation de ce programme, mais on décrira d'abord dans le chapitre suivant les techniques d'analyse, de synthèse et de reconnaissance de la parole. Car ces techniques sont particulières au dialogue oral et conditionnent au moins en partie tout module du projet dont fait partie le programme de génération de phrases.

Cependant, il faudrait aussi fournir les énoncés sémantiques au programme de génération automatiquement en fonction du contexte du dialogue, alors qu'on s'est contenté de les donner à la main (sur carte ou par clavier). Cela exigerait de réaliser un module sémantico-pragmatique capable de raisonner au cours du dialogue, pour générer les énoncés sémantiques et pour aider la reconnaissance de parole (en prédisant les réponses possibles). En fait ce module intelligent reste encore à développer, bien qu'on ait déjà réalisé des dialogues rudimentaires simulant par exemple un opérateur téléphonique.

II - TRAITEMENT DE LA PAROLE

La parole est une activité complexe et elle peut être étudiée à plusieurs niveaux différents du psychologique au physique.

De fait, dans le dialogue on passe successivement par ces différents niveaux lors de la production et de la reconnaissance de la parole par les locuteurs. Au niveau psychologique ou sémantique la parole permet de communiquer des significations, d'une façon que l'on ne sait d'ailleurs pas encore très bien comment décrire. Au niveau linguistique par contre l'information à transmettre est encodée selon des règles syntaxiques, lexicales, morphologiques, et phonologiques que l'on commence à bien connaître. Puis au niveau articulatoire, la structure abstraite du langage est transformée en mouvements des muscles de l'appareil phonatoire, commandés par des impulsions nerveuses venant du cerveau. Enfin ces mouvements font vibrer l'air ambiant et les sons ainsi émis se propagent autour du locuteur. On est alors au niveau physique, ou acoustique proprement dit.

Lorsque ces vibrations atteignent les oreilles de l'auditeur, un processus inverse a lieu et on repasse les niveaux dans l'autre sens. Les impulsions nerveuses envoyées par l'oreille au cerveau sont organisées et décodées en unités phonologiques qui sont elles-mêmes regroupées en unités morphologiques et lexicales, puis syntaxiques. Enfin, à ces représentations linguistiques sont associées des significations plus ou moins proches de celles que voulait transmettre le locuteur lorsqu'il a prononcé ces sons du langage. Il est très probable que le contexte du dialogue influe beaucoup sur les significations associées aux paroles prononcées, en plus de la forme strictement linguistique du message.

Ceci dit, s'il est certain qu'il faille distinguer dans la structure abstraite de la langue ces différents niveaux hiérarchisés, il est possible que l'encodage de la parole par le locuteur et son décodage par l'auditeur fasse intervenir les différents niveaux en même temps et non pas de façon rigoureusement séquentielle. Par exemple les travaux effectués en reconnaissance automatique de la parole amènent à penser que la perception des phonèmes est en partie conditionnée par la connaissance des mots et des

formes syntaxiques possibles dans une langue donnée ainsi que par le sens général du contexte. De même, il est probable que l'expression de la pensée s'adapte au répertoire des structures syntaxiques et du vocabulaire disponible au locuteur. Il y aurait alors interaction entre les différents niveaux du langage, et il faudrait supposer une organisation parallèle dans l'utilisation du langage.

Au niveau acoustique, c'est-à-dire dès que les sons articulés par le locuteur ont franchi ses lèvres et avant d'être perçue par l'oreille d'un auditeur, la parole peut être considérée comme un phénomène purement physique et traitée comme n'importe quel mouvement vibratoire. Les sons de la parole n'étant à ce niveau que des variations de la pression de l'air, il est possible d'en donner une description physique exacte. En pratique cela a été permis essentiellement par le développement des appareils électroniques de mesure puis par l'informatique.

On commence par convertir les vibrations sonores en un signal électrique analogue grâce à des capteurs comme les microphones. L'appareillage dont on dispose actuellement permet alors d'appliquer à ce signal toute une panoplie de techniques de traitement du signal, et en particulier l'analyse spectrale de Fourier. On effectue ainsi une analyse acoustique de la parole, et les résultats de l'analyse sont ensuite utilisés pour synthétiser la parole, ce qui permet de vérifier l'analyse. Enfin grâce aux connaissances acquises en analyse et synthèse on peut tenter la reconnaissance automatique de la parole. On aboutit à un traitement automatique de la parole qui simule la production et la reconnaissance.

1) ANALYSE

L'analyse est bien sûr la base de départ de tout traitement. Il faut d'abord arriver à décrire l'objet des recherches. Or on ne peut pas dire grand chose du signal de parole brut tel qu'il se présente avant l'analyse, sur l'écran d'un oscilloscope par exemple. C'est à peine si on peut distinguer les consonnes des voyelles dans la forme d'onde. Pour faire des distinctions plus fines on a recours le plus souvent à l'analyse spectrale de Fourier.

1-1 Analyse spectrale

L'analyse spectrale est en effet fondamentale en traitement du signal. Selon le théorème de Fourier, tout signal périodique quelle que soit sa forme d'onde peut se décomposer en une somme de sinusoides simples harmoniques

dite série de Fourier :

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \sin(n\omega_0 t + \varphi_n)$$

$f(t)$ étant la fonction périodique représentant le signal par rapport au temps t ; a_0 la composante continue ; n le numéro (entier) de l'harmonique ; ω_0 la pulsation égale à 2π fois la fréquence fondamentale ; a_n l'amplitude de chaque harmonique ; et φ_n sa phase.

La décomposition en série de Fourier permet donc de représenter toute vibration périodique par un spectre de raies harmoniques en donnant une amplitude et une phase à chaque composante sinusoïdale.

On peut aussi écrire la série de Fourier sous une autre forme, équivalente à la notation trigonométrique mais plus compacte, la notation exponentielle complexe :

$$f(t) = \sum_{n=-\infty}^{\infty} \kappa_n e^{jn\omega_0 t}$$

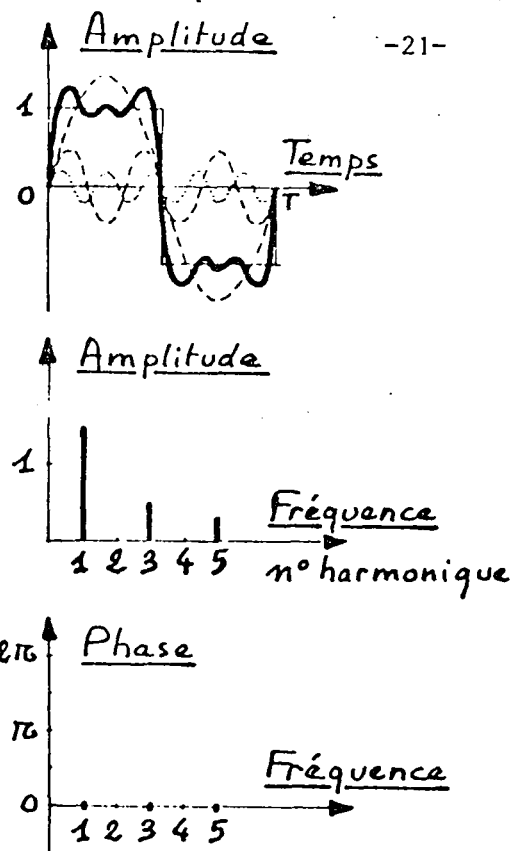
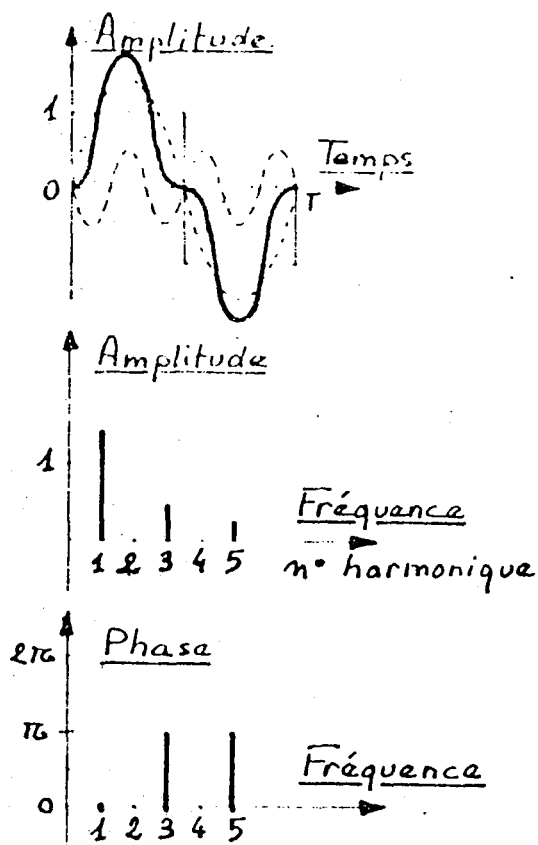
κ_n étant un coefficient complexe correspondant à l'amplitude et à la phase des composantes de la notation sinusoïdale.

L'analyse spectrale se généralise aux signaux non-périodiques comme les bruits. En considérant que la période tend alors vers l'infini, les raies du spectre se resserrent et la série devient une somme continue de sinusoïdes. Cette analyse continue s'exprime par l'intégrale de Fourier :

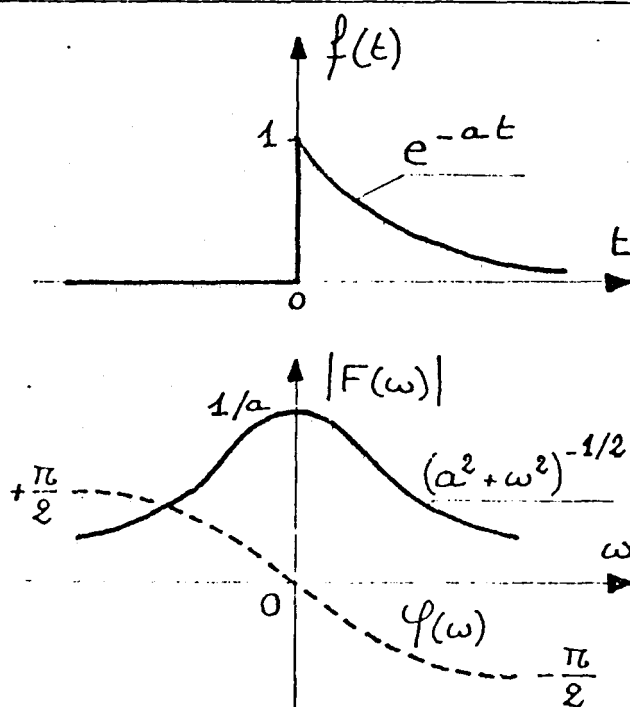
$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega$$

ω étant la pulsation, et $F(\omega)$ la transformée de Fourier de la fonction $f(t)$. La transformée de Fourier est l'équivalent continu des coefficients de la série de Fourier, et s'exprime par la relation complexe :

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$

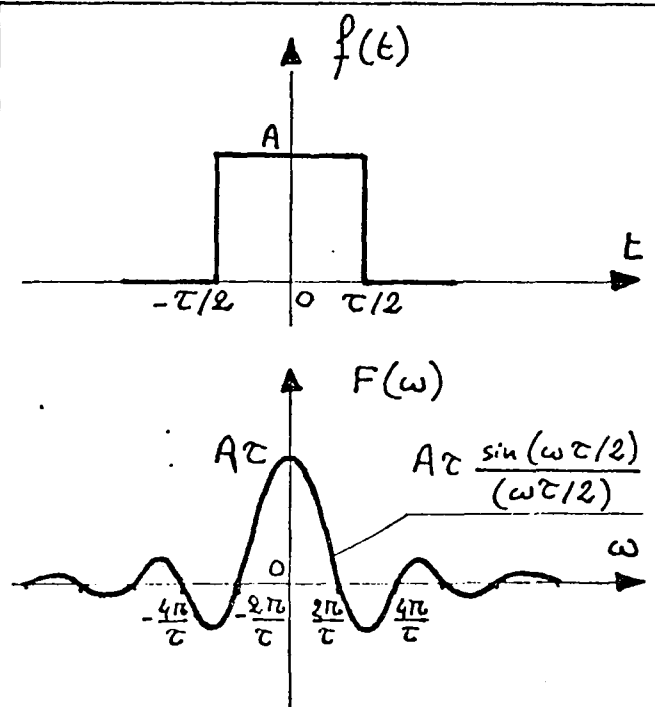


Représentation temporelle de deux signaux ayant la même composition harmonique, mais des spectres de phase différents. La figure de droite représente un signal carré approché, reconstitué à partir de ses trois premiers harmoniques.



Transformée de FOURIER d'un signal exponentiel. Le spectre d'amplitude $F(\omega)$ est complexe :

$$F(\omega) = |F(\omega)| \cdot e^{-j\phi(\omega)}$$



Transformée de FOURIER d'une impulsion carrée. La fonction est paire; le spectre de phase est identiquement nul, et le spectre d'amplitude est réel.

.../...

La transformée de Fourier donnant l'amplitude et la phase en fonction de la fréquence permet donc en général de représenter un signal quelconque par un spectre continu. On obtient ainsi une représentation strictement équivalente au signal temporel, mais dans le domaine fréquentiel.

En fait, pour être tout à fait rigoureux il faut préciser que la transformée de Fourier n'est pas toujours possible, en particulier si le signal est illimité dans le temps. Mais on peut alors analyser une durée finie du signal grâce à une relation très similaire, la transformée de Laplace :

$$F(p) = \int_0^{\infty} f(t) e^{-pt} dt$$

avec $p = \alpha + j\omega$ ce qui revient à calculer la transformée de Fourier du signal fini $f(t) e^{-\alpha t}$. Quant aux limitations de la transformée de Laplace, elles ne posent pas de problèmes en pratique.

Enfin, à cause de l'utilisation croissante des appareils digitaux et du calcul numérique sur ordinateur, il faut mentionner la transformée de Fourier discrète. Après échantillonnage (à plus de 2 fois la fréquence maximale du signal) et conversion analogique-numérique, elle permet de calculer le spectre digital du signal numérisé :

$$F(m) = \sum_{n=0}^{N-1} f(n) e^{-jmn 2\pi/N}$$

$m = 0, 1, \dots, N-1$ étant les numéros des échantillons de fréquence ; n les numéros des échantillons de temps ; et N le nombre des échantillons de fréquence ou de temps.

Par ailleurs, on s'intéresse en général à l'évolution du signal au cours du temps. Il faut donc pouvoir décrire l'évolution correspondante du spectre dans le temps. Pour cela on considère non pas le spectre du signal dans son ensemble, mais plutôt une suite de spectres dits "instantanés" calculés sur des intervalles successifs du signal. Ces intervalles sont prélevés à travers une "fenêtre" temporelle dont la forme et la durée conditionnent le spectre obtenu. En particulier on ne peut pas améliorer à la fois la résolution en fréquence et la résolution en temps du fait de la relation d'incertitude :

$$\Delta f \cdot \Delta t \simeq 1$$

Ainsi une fenêtre brève donne une bonne résolution temporelle mais une mauvaise définition fréquentielle, alors qu'une fenêtre allongée améliore la précision en fréquence mais amortit les variations temporelles.

.../...

.../...

C'est une limitation fondamentale de l'analyse.

En résumé, l'analyse spectrale permet de décomposer un signal quelconque selon trois dimensions : l'amplitude, la fréquence et le temps. Car le plus souvent on néglige la phase. On obtient donc la distribution de l'amplitude (ou de l'énergie) selon la fréquence, et l'évolution de cette distribution spectrale dans le temps.

L'intérêt d'un tel traitement est bien sûr de réduire une vibration aussi complexe soit-elle à quelques paramètres simples. Cette analyse permet d'appréhender plus clairement des phénomènes qui ne présentent à première vue pas d'organisation nettement discernable. C'est une démarche tout à fait générale, mais qui est particulièrement précieuse dans le cas du signal de parole étant donné sa complexité et sa richesse en information.

Ensuite il est certain que l'oreille humaine effectue une analyse fréquentielle similaire sur les sons. En première approximation on peut dire que l'oreille perçoit le spectre d'amplitude et non la forme d'onde du son. Les variations de phase n'étant pas identifiées, deux sons ayant des formes temporelles très différentes seront perçus comme identiques s'ils ont un spectre d'amplitude semblable, quelles que soient les phases. Une représentation par spectre d'amplitude, d'énergie ou de puissance est donc plus conforme à la perception, ce qui se semble se vérifier en pratique.

Cela dit, l'analyse spectrale de Fourier n'est pas la seule analyse possible, même si elle est la plus courante. C'est une description mathématique très commode, mais qui n'est pas inhérente au phénomène physique envisagé. Parmi d'autres méthodes utilisables on peut citer en particulier la mesure des passages à zéro et le codage prédictif. La méthode des passages à zéro consiste à retenir les points où s'annulent le signal et ses dérivées successives. C'est une méthode qui a été assez souvent employée, car facile à réaliser en analogique ou en digital.

Le codage prédictif linéaire (ou "LPC" : "Linear Predictive Coding") extrait des paramètres du signal par modélisation. Les paramètres d'un modèle linéaire sont ajustés en minimisant l'écart entre le signal observé et le signal prédit par le modèle en fonction des échantillons précédents. Cette méthode récente se révèle très efficace malgré le volume des calculs nécessaires et on commence à l'employer de plus en plus.

.../...

.../...

On peut cependant considérer ces diverses méthodes comme à peu près équivalentes. La démarche est fondamentalement la même : on veut décrire un phénomène complexe et variable au moyen d'un petit nombre de paramètres simples. Il est d'ailleurs possible de passer d'une description à une autre. Par exemple, on peut très bien calculer le spectre fréquentiel à partir des coefficients de prédiction linéaire, et les passages à zéro renseignent sur les fréquences composantes. Le plus souvent on utilise telle ou telle méthode selon la commodité et les moyens disponibles. Mais il faut en tous cas rappeler cette évidence si souvent méconnue que les résultats de l'analyse dépendent de la méthode employée.

1-2 Moyens d'analyse

Les formules mathématiques qu'on vient de voir indiquent comment on pourrait théoriquement calculer le spectre d'un signal. Mais comme le signal de parole évolue sans cesse, les difficultés analytiques et la lourdeur des calculs nécessaires font qu'en pratique on doit utiliser d'autres méthodes plus rapides. L'analyse mathématique doit être considérée comme la théorie sous-jacente à l'utilisation de moyens plus pratiques et non comme une méthode de mesure. Concrètement on a donc souvent recours à des filtres électroniques divers, ou bien encore au calcul numérique par ordinateur.

. Filtres électroniques

Dès le 19ème siècle on a utilisé des résonateurs mécaniques pour analyser les voyelles. Mais c'est essentiellement le développement de l'électronique qui a permis des analyses précises, grâce à l'emploi de circuits résonnants comme filtres.

Le moyen le plus rapide d'analyse consiste donc à utiliser un banc de filtres électroniques. Chaque filtre répond de façon privilégiée à une bande de fréquences, et en étageant les filtres d'après leur fréquence de réponse on obtient à la sortie du banc de filtres le spectre du signal d'entrée. La réponse est immédiate, continue, et suit fidèlement l'évolution du signal d'entrée. On retrouve tous les avantages du calcul analogique lorsqu'il faut effectuer des opérations complexes en temps réel : les filtres restent le moyen d'analyse le plus rapide et le plus simple.

.../...

Par exemple l'analyseur qui constitue le premier maillon de la chaîne de reconnaissance de la parole au LIMSI est un banc de 32 filtres, réglables chacun en fréquence, gain et bande passante. Car il ne faut pas oublier que le spectre obtenu dépend des caractéristiques des filtres. La finesse du spectre en fréquence croît évidemment avec le nombre de filtres sur la bande considérée et avec la finesse de résolution fréquentielle de chacun des filtres. Et comme on a vu, la précision fréquentielle dépend de la "fenêtre temporelle" utilisée : la résolution en fréquence est inversement proportionnelle à la résolution temporelle, c'est-à-dire au temps de réponse du filtre. On ne peut pas avoir plus de précision dans les deux domaines à la fois.

Pour contrôler le spectre obtenu il est donc absolument nécessaire de pouvoir modifier les caractéristiques des filtres selon les besoins. C'est possible dans certaines limites. Mais les capacités d'adaptation du matériel analogique sont limitées de par sa construction même et on peut être amené à recourir au digital pour la grande souplesse des filtres numériques qu'on obtient par simulation sur ordinateur. Un filtre numérique est simplement un programme de calcul qui à partir des échantillons numériques du signal fournit les nouvelles valeurs du signal filtré. On peut en faire varier tous les paramètres beaucoup plus facilement qu'avec un filtre matériel, mais au prix d'un certain temps de calcul. C'est une méthode de plus en plus utilisée, car le logiciel numérique est plus maniable que le matériel analogique.

. Sonagraphe

Le sonagraphe n'est qu'un exemple particulier d'analyseur à filtre. Mais il a été le premier appareil commercial vraiment utilisable (il est sorti à la fin de la 2ème guerre mondiale) et sa relative simplicité d'emploi fait qu'il reste l'analyseur le plus répandu en recherche acoustique, malgré son âge.

Il se compose essentiellement non d'un banc de filtres, mais d'un seul filtre fixe. Le spectre est alors exploré grâce à un procédé hétérodyne c'est-à-dire des circuits qui font varier graduellement la fréquence du signal d'entrée de façon à faire défiler tout son spectre devant la bande passante du filtre. On obtient bien le spectre d'intensité, mais pas en temps réel, puisque ce système équivaut à l'emploi d'un filtre variable qui explorerait successivement les fréquences du spectre (alors qu'un banc de filtres analyse toute l'étendue du spectre en même temps).



SONAGRAMME EN BANDE ETROITE DE LA PHRASE
" JE DORMAIS SOUS UN ABRICOTIER "



SONAGRAMME EN BANDE LARGE DE LA MEME PHRASE

Il faut donc un temps d'analyse et l'appareil ne peut travailler que sur des enregistrements. Pour les 2, 4 secondes de signal que le sonographe est capable de mémoriser sur piste magnétique, il lui faudra plusieurs minutes pour en faire l'analyse. C'est un gros défaut de cet appareil. Par contre la représentation du résultat sur papier, où l'intensité est marquée de façon approximative mais suffisante par le noircissement, est très commode à utiliser. Et l'appareil permet de nombreux réglages entre autres celui de la largeur de la bande du filtre, et donc inversement de son temps de réponse. En bande étroite on distingue les harmoniques au détriment de la précision temporelle. Par contre en bande large on n'a que l'enveloppe du spectre mais on distingue bien les phénomènes brefs.

Certes, le sonographe est un analyseur assez rustique comparé aux moyens plus modernes mais il se montre souvent suffisant en pratique et il reste encore très utilisé. Ce qui fait que la représentation acoustique de la parole la plus souvent publiée consiste en "sonagrammes" obtenus avec cet appareil : le spectre est donné avec le temps en abscisse, la fréquence en ordonnée, et l'intensité est représentée par le noircissement du papier. Le sonographe doit donc être considéré comme un outil de base.

. Calcul numérique

Il n'est pas possible en général de calculer à la main une analyse spectrale. Mais on peut utiliser des méthodes numériques calculables par ordinateur. La transformée de Fourier discrète indique comment obtenir le spectre numérique d'un signal préalablement échantillonné et quantifié. L'algorithme de calcul le plus répandu est la transformée de Fourier rapide (ou "FFT" : Fast Fourier Transform) qui permet de réduire considérablement le temps de calcul. Cependant, malgré l'amélioration constante des algorithmes, les calculs numériques restent lourds, et la FFT n'est pas assez rapide pour atteindre le temps réel, c'est-à-dire l'analyse de la parole au fur et à mesure qu'elle est prononcée.

C'est aussi le calcul par ordinateur qui permet le filtrage numérique et la mise en oeuvre de diverses méthodes d'analyse comme le codage prédictif. La précision et la souplesse des méthodes digitales expliquent leur succès croissant. Elles remplacent de plus en plus le matériel analogique, même si elles restent encore souvent relativement lentes. En permettant d'expérimenter facilement de nouvelles méthodes, le calcul numérique a fait évoluer le traitement du signal. Et une fois les

algorithmes mis au point, on peut employer des processeurs spécialisés pour augmenter la vitesse de calcul. Mais il faut toujours tenir compte des problèmes d'échantillonnage et de conversion entre numérique et analogique, en plus des problèmes propres au calcul numérique.

Enfin, pour conclure cette revue des moyens techniques d'analyse on doit ajouter une restriction. Il faut bien voir que la description physique n'est pas une fin en soi. Le signal de parole est un phénomène linguistique qui présente une structure très particulière. A partir d'une quantité énorme d'information acoustique, l'auditeur extrait les paramètres significatifs de l'information proprement linguistique, qui est bien moindre.

La redondance est très grande à cause des contraintes hiérarchisées (phonologie, lexique, syntaxe, sémantique) de la structure linguistique. En fin de compte, ce qui importe vraiment, c'est la façon dont le signal acoustique est perçu et interprété. La description doit être pertinente, et ne retenir que les éléments fonctionnels du message, sous peine de s'égarer dans une masse d'informations inutiles.

1-3 Phonétique acoustique

Ces moyens pratiques d'analyse de la parole ont permis d'élaborer une phonétique acoustique. Cette phonétique instrumentale donne une description des sons du langage plus précise et plus efficace que la phonétique traditionnelle articulatoire ou perceptive. Car seule une description physique exacte permet la synthèse ou la reconnaissance automatique de la parole. Ceci dit, les différentes méthodes de description se recoupent en grande partie, et de toutes façons on doit considérer aussi la fonction linguistique des faits physiques.

Ainsi parmi les nombreuses publications dans ce domaine après la dernière guerre mondiale, la plus marquante a sans doute été "Preliminaries to Speech Analysis" écrite en 1952 par un ingénieur (Fant) et deux linguistes (Halle et Jakobson). Les traits distinctifs des phonèmes y sont systématiquement redéfinis en termes acoustiques. A la même époque Delattre, Cooper, Liberman (entre autres) précisaient les traits physiques pertinents par analyse puis synthèse. En bref, les progrès de la technique ont renouvelé la phonétique en la transformant en science expérimentale.

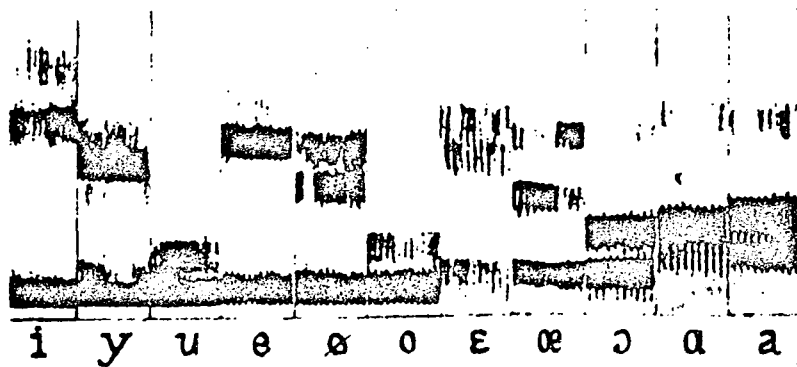
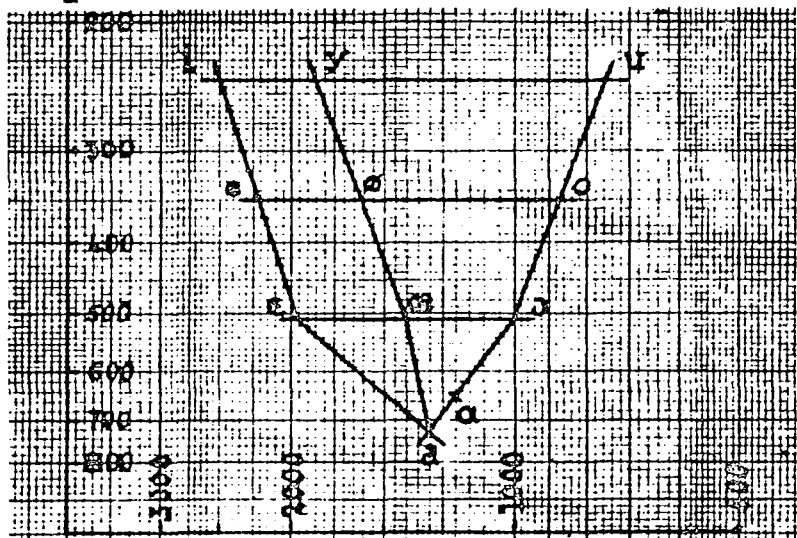
Du point de vue acoustique, on considère donc maintenant le conduit vocal comme un tuyau sonore ouvert à un bout. Une telle cavité donne lieu à des ondes stationnaires pour des fréquences déterminées. Ces fréquences de résonance dépendent de la longueur et de la forme du tuyau, et varient donc avec l'articulation. Comme le conduit vocal est assez amorti, on observe plutôt des zones de résonance : les formants. On peut alors décrire les sons de la parole comme le produit d'une source (ton glottal, ou bruits d'écoulement) et d'un filtre, le conduit vocal.

La source peut être harmonique (cordes vocales) ou bruyante (constrictions du conduit vocal) ; ces deux types de source peuvent d'ailleurs fonctionner simultanément (consonnes voisées). L'énergie fournie par la source est modifiée par les résonances du filtre que constitue une configuration donnée du conduit vocal. C'est surtout ce filtre qui détermine la forme du spectre final, et les formants ne dépendent que de l'articulation, c'est-à-dire de la position de la langue et des lèvres avec éventuellement l'ouverture des fosses nasales en parallèle.

Il est ainsi possible de donner une description phonétique en termes acoustiques, en spécifiant la source et les formants. Concrètement c'est grâce à l'analyse spectrale et l'expérimentation avec la parole de synthèse que s'est élaboré cette phonétique acoustique. En gros on peut distinguer de cette façon entre les sons vocaliques ayant une structure de formants nette, et les sons consonantiques qui sont plutôt des bruits.

Les voyelles se différencient par la position de leurs formants caractéristiques : le premier formant s'élève avec l'ouverture de la langue, les formants suivants s'abaissent pour les voyelles postérieures. Les nasales et liquides montrent aussi des formants, mais plus faibles. De même les bruits consonantiques se distinguent par leur zone de résonance, plus aigüe pour les dentales par exemple. Mais les occlusives sont surtout identifiées grâce aux transitions caractéristiques des formants vocaliques adjacents, alors que le bruit propre des fricatives est plus net.

Les transitions des voyelles adjacentes correspondent donc au lieu d'articulation des consonnes. Enfin, le ton glottal, normalement toujours présent lors des sons vocaliques, se superpose au bruit des consonnes si elles sont voisées.



Voyelles du Français

LABIAL

DENTAL

PALATOALVEOLAR

PALATAL

VELAR

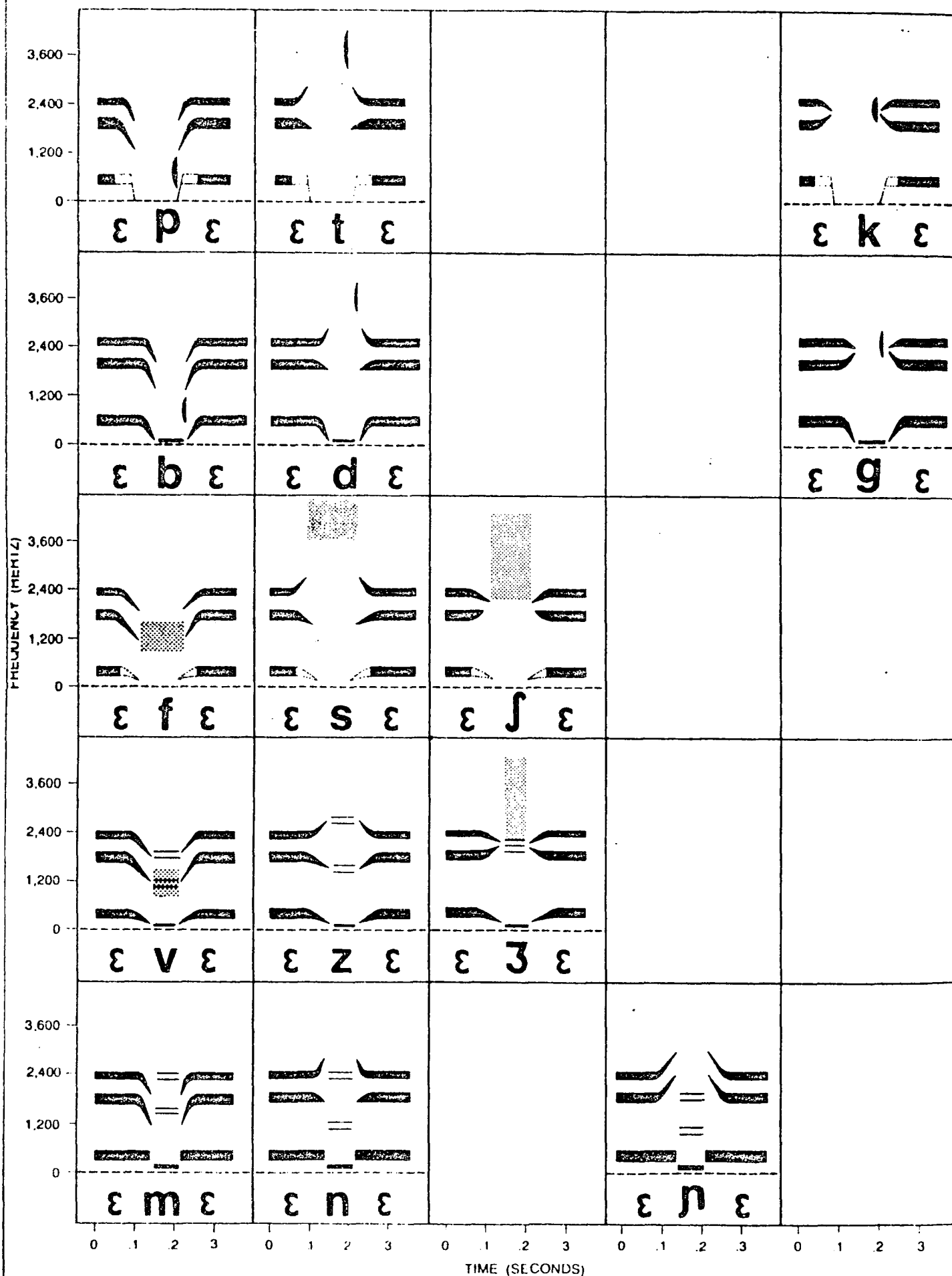


Schéma des consonnes du Français

.../...

Cette rapide nomenclature appelle cependant plusieurs remarques supplémentaires. On voit d'abord que cette classification acoustique reste en grande partie parallèle à la classification articulatoire traditionnelle qui a en fait servi de fil directeur. Cela se justifie en constatant que l'articulation détermine la forme sonore. Et comme l'articulation est mouvement continu, les sons produits varient aussi en un flot continu, qu'il est difficile de segmenter en unités phonémiques. De même il est très difficile d'identifier précisément la réalisation physique des traits distinctifs.

Car en général les éléments influent les uns sur les autres et varient selon le contexte, et selon le locuteur. On ne peut donner que des valeurs relatives, et de plus les indices acoustiques sont souvent complexes et ambigus. Par exemple le voisement se manifeste tout autant par une durée et une intensité réduite de la consonne que par la présence du ton glottal. La correspondance entre les unités fonctionnelles et leur réalisation physique est donc plus compliquée qu'on ne le croyait. D'où les difficultés rencontrées en synthèse et en reconnaissance de la parole.

2) SYNTHESE

Les moyens que l'on vient de voir permettent une analyse acoustique détaillée de la parole. On peut alors utiliser les connaissances ainsi obtenues sur la structure du signal parlé pour tenter d'en faire la synthèse. Avec un appareillage approprié on produira ainsi de la parole artificielle. Outre l'intérêt en soi d'une telle réalisation, la synthèse permet de vérifier les résultats de l'analyse. Si la parole produite est bien intelligible, c'est qu'on a extrait de l'analyse les paramètres pertinents—ce dont on ne pouvait être sûr sans synthèse. La synthèse est donc un complément indispensable à l'analyse de la parole. Mais pour cela, il faut des appareils appropriés : les synthétiseurs.

Les premiers essais connus de synthèse de parole, à partir du 18^{ème} siècle, utilisèrent d'abord des résonateurs mécaniques, plus ou moins analogues du conduit vocal. Mais ils étaient difficilement perfectibles et on n'utilise plus que des synthétiseurs électroniques, souvent commandés par ordinateur, qui sont beaucoup plus souples et précis.

.../...

2-1 Synthétiseurs articulatoires

On peut distinguer deux classes principales de synthétiseurs électroniques : les analogues articulatoires et les analogues acoustiques (mais la terminologie varie beaucoup). Les analogues articulatoires simulent la forme du conduit vocal, considéré comme une suite de tubes de longueur et de section variables. Chaque tube présente une impédance acoustique que l'on représente par un circuit électrique résonant d'impédance analogue.

En ajustant les paramètres des circuits de façon à figurer l'articulation et en excitant le montage par une source périodique ou de bruit on produit de la parole continue.

Ces synthétiseurs mettent directement en évidence des contraintes mécaniques dues au fonctionnement réel du conduit vocal. Mais ils sont difficiles à mettre en oeuvre parce que les paramètres articulatoires sont souvent mal définis. Pour plus de précision il faut recourir au procédé peu commode de la cinématographie aux rayons X des mouvements articulatoires, ou bien calculer la fonction d'aire du conduit vocal à partir du signal acoustique. Malgré les connaissances que ces synthétiseurs apportent sur l'articulation des sons, ils sont donc encore relativement peu employés.

2-2 Synthétiseurs acoustiques

Par contre, les analogues acoustiques sont les plus répandus. Ceux-ci essaient uniquement de reproduire le signal acoustique à partir de son spectre, sans faire référence à l'articulation. Comme c'est la structure acoustique qu'on essaye alors de représenter, on est cette fois plus proche de l'aspect auditif de la parole puisqu'on a vu que l'oreille effectue une analyse spectrale semblable. En gros, ces synthétiseurs acoustiques se composent d'une série de générateurs variables disposés de façon à couvrir le spectre utile du signal de parole, et dont on mélange les sorties pour reconstituer le signal global. L'agencement de ces générateurs varie selon la conception choisie.

. Synthétiseurs à canaux

On peut en effet répartir les analogues acoustiques en deux types différents : les synthétiseurs à canaux et les synthétiseurs à formants, plus évolués. Les synthétiseurs à canaux découpent le spectre en bandes ou "canaux" échelonnés régulièrement en fréquence. Les canaux sont réalisés

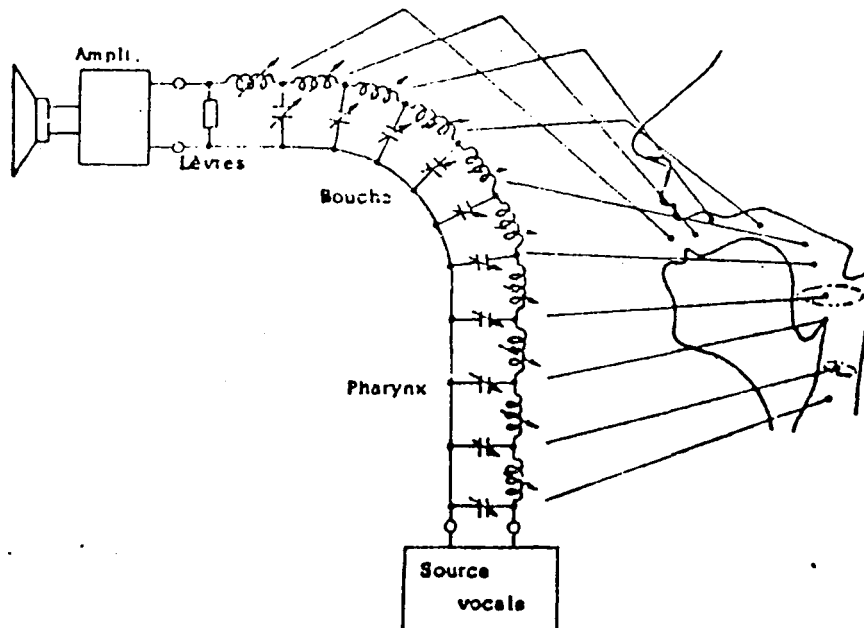
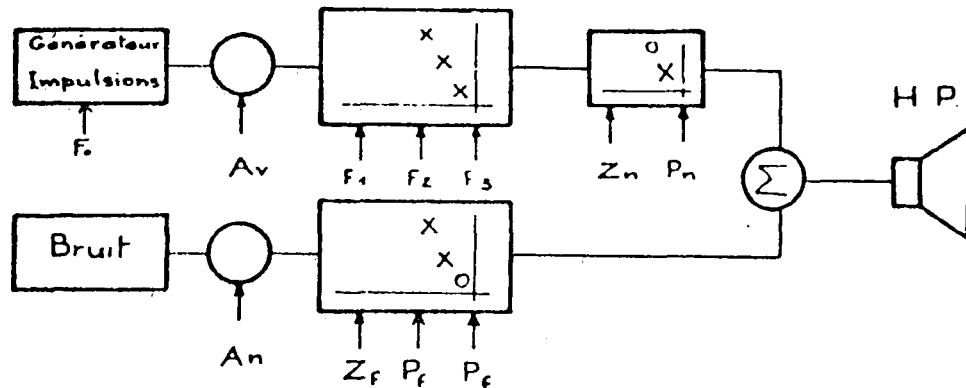


Schéma du Synthétiseur analogue du conduit Vocal LEA de PANT



- Principe d'un synthétiseur paramétrique digital (d'après Flanagan)
- F_0 et A_v : fréquence et amplitude du spectre de raies
- F_1, F_2, F_3 : fréquence des trois premiers formants
- Z_n, P_n : caractéristiques spectrales des semi-vowelles
- A_n : amplitude du spectre continu (bruit)
- Z_f, P_f, P_f : caractéristiques spectrales des consonnes

Types de Synthétiseurs

.../...

par un banc d'oscillateurs, ou un banc de filtres attaqués par une source de voix ou de bruit. Chaque canal est commandé indépendamment de manière à reproduire le spectre de n'importe quel son. Le principe est simple et efficace et ces synthétiseurs ont été les premiers utilisés, en particulier dans les systèmes de compression de bande comme le Vocoder. Ils sont maintenant très au point et la parole produite est parfaitement intelligible.

La commande peut être optique, grâce à des cellules photoélectriques disposées de façon à pouvoir relire un sonagramme, chaque cellule étant affectée à une bande de fréquence. Ces sélecteurs de sonagrammes ("Pattern-Playback") permettent de tester l'intelligibilité de sonagrammes schématiques dessinés à la main. Cette technique commode a fourni un outil de base pour les recherches fondamentales sur la structure acoustique de la parole. Par contre le procédé reste artisanal et n'est évidemment pas adapté à une utilisation intensive. De fait la commande des synthétiseurs se fait en général par ordinateur.

Par exemple le synthétiseur à canaux "Icophone IV" du LIMSI se compose d'une batterie de 44 oscillateurs sinusoïdaux échelonnés tous les 100 Hertz en parallèle. On couvre donc un spectre de 100 à 4 400 Hz. Ces 44 composantes sont commandées en tout ou rien par l'ordinateur puis mélangées avant amplification finale. De plus on peut faire varier la fréquence de tous les oscillateurs harmoniques en même temps de façon à commander l'intonation. Car en modifiant solidairement la fréquence de chaque harmonique, on change la fréquence fondamentale. Il y a aussi trois générateurs de bruit pour la synthèse des sons sourds. Les paramètres de commande sont stockés en fichiers sur mémoire de masse et la synthèse est assurée par un programme de gestion.

. Synthétiseurs à formants

Mais il faut remarquer que le découpage en canaux ne correspond pas à la répartition caractéristique de l'énergie dans un spectre de parole, qui peut se définir par un petit nombre de résonances particulières appelées "formants". Faute de tenir compte de la structure des sons à reproduire, un synthétiseur à canaux nécessite donc un nombre trop élevé de paramètres de commande, sans rapport avec les traits réellement pertinents.

.../...

Par contre, les synthétiseurs paramétriques à formants ne reproduisent que les formants caractéristiques et les paramètres nécessaires sont moins nombreux et plus intéressants. On constate d'ailleurs une certaine correspondance avec les paramètres articulatoires de la phonétique classique, ce qui montre encore la plus grande pertinence linguistique de ces synthétiseurs à formants.

Les formants sont reproduits par trois ou quatre filtres variables qui sont attaqués par une source de bruit (pour les sons sourds) et une source de voix (pour les sons voisés). Les filtres peuvent être en parallèle ou en série selon les modèles, et s'ils sont montés en parallèle, il faut en mélanger les sorties pour reconstituer le spectre entier. Les paramètres de commande sont les caractéristiques des filtres, qui correspondent à la fréquence et à l'intensité des formants, et les caractéristiques de la source comme la fréquence fondamentale, qui correspond à l'intonation.

Si ces paramètres sont bien choisis, l'intelligibilité peut être excellente, et la commande par ordinateur permet la production de parole continue en temps réel. En conclusion, on doit considérer les synthétiseurs à formants comme les plus utilisables des appareils simulant la fonction de transfert du conduit vocal.

Par exemple, le synthétiseur "Icophone VI" du LIMSI se compose d'un générateur de bruit et d'un générateur de voix (périodique) qui attaquent quatre filtres passe-bande en parallèle. Les générateurs simulent ainsi la source d'énergie excitatrice et les filtres les fréquences de résonance du conduit vocal, c'est-à-dire les formants. Les paramètres de commande, délivrés sous forme numérique directement par l'ordinateur, sont au nombre d'une quinzaine. Ce sont essentiellement les amplitudes des deux générateurs, la fréquence du générateur périodique (c'est-à-dire l'intonation) et les fréquences centrales, bandes passantes et gains respectifs de chacun des quatre filtres (c'est-à-dire les formants). C'est cet appareil qui est actuellement le plus utilisé pour les recherches du laboratoire.

résumer
X
Cependant, comme pour l'analyse et pour les mêmes raisons de souplesse d'emploi, on tend actuellement en synthèse à recouvrir aux méthodes numériques. Au lieu d'utiliser des circuits analogiques relativement figés, on peut les simuler en numérique, ou même calculer directement la forme d'onde. Le calcul numérique permet ainsi de faire varier plus librement les paramètres de synthèse. En particulier on obtient une excellente

.../...

synthèse à partir des coefficients de prédiction linéaire. En synthèse aussi on assiste donc à une utilisation croissante des méthodes numériques, sur calculateur universel ou grâce à des circuits digitaux spécialisés.

. Commandes de la synthèse

Enfin quel que soit le synthétiseur utilisé, il faut aussi pouvoir le commander. On a dit que la commande se fait en général par ordinateur, mais les méthodes diffèrent selon les unités choisies pour la synthèse. Comme la parole est physiquement continue et que les éléments phonétiques varient beaucoup avec le contexte, il ne suffit pas d'accoler les phonèmes les uns à la suite des autres. Il faut prévoir les transitions entre ces éléments pour chaque paramètre.

On pourrait résoudre le problème en gardant en mémoire des unités aussi grandes que des mots. C'est la méthode du genre de l'"horloge parlante". Il n'y a plus qu'à arranger la prosodie, mais le vocabulaire est limité et doit être enregistré à l'avance : c'est un système figé. Par contre si on utilise un nombre réduit d'unités plus petites comme les phonèmes, la synthèse est illimitée avec un faible encombrement de la mémoire, mais il faut des règles de calcul des transitions. C'est la "synthèse par règles", qui permet de prononcer n'importe quelle phrase. Mais des règles assez fines peuvent demander un certain temps de calcul.

Une autre méthode plus simple est la synthèse par "diphonèmes", employée au LIMSI. On garde alors en mémoire les paramètres pour chaque combinaison de deux phonèmes, considérée comme une unité de base. Ces diphonèmes sont associés par leur partie stable pour reconstituer une phrase. Ainsi le mot "unité" est synthétisé avec /yn+ni+it+te/. Il suffit de 600 diphonèmes environ pour le français. C'est un nombre assez important d'éléments à mettre au point et à stocker, même si on optimise le rangement en mémoire. Mais le problème des transitions est ainsi résolu sans peine et la synthèse est très rapide.

.../...

2-3 Synthèse à partir de texte

Pour obtenir la prononciation d'une phrase par un synthétiseur, on doit le commander en code phonétique. Mais il est beaucoup plus agréable de pouvoir donner les phrases à prononcer en orthographe ordinaire dont on a l'habitude. Il est également utile de pouvoir fournir directement au synthétiseur du texte imprimé. Pour cela il faut avoir un programme de traduction automatique de l'orthographe en phonétique. Mais cette conversion graphèmes-phonèmes présente de nombreux problèmes.

En effet, l'orthographe est une norme culturelle qui ne reflète que plus ou moins la prononciation, au point que l'apprentissage d'une orthographe "correcte" demande plusieurs années scolaires. La langue écrite ne s'utilise pas dans les mêmes conditions que la langue parlée et en diffère donc toujours plus ou moins. Pour assurer la communication à distance dans le temps et dans l'espace, sans contexte de situation, la langue écrite est plus conservatrice et plus stricte. Ainsi l'orthographe correspond souvent à un état antérieur de la prononciation agrémenté de considérations étymologiques.

D'où le maintien graphique de l'unité des morphèmes dans les familles de mots (comparer "lent, lente, lenteur" avec "grand, grande, grandir", "plein, pleine" avec "fin, fine, finir", ou bien en anglais "type, typical", "sane, sanity"). D'où aussi une tendance quasi-idéographique à distinguer les homophones ("sot, saut, seau, sceau", "fin, faim", ou en anglais "right, write, rite"). Par contre la prosodie est très mal indiquée.

La situation varie d'ailleurs d'une langue à l'autre. L'orthographe de l'espagnol est plus proche de la prononciation que celle du français, et l'orthographe anglaise en est encore plus éloignée ; l'écriture du chinois n'a que peu de rapport avec la prononciation. Les difficultés de la conversion en phonétique varient bien sûr en conséquence.

En tous cas il faut à la fois des règles de conversion et un dictionnaire d'exceptions. Les exceptions, à la prononciation imprévisible ("monsieur", "oignon"), sont en plus ou moins grand nombre selon la langue. Par exemple en anglais le dictionnaire doit être très étendu. On peut alors avoir intérêt à décomposer les mots en morphèmes, dont le nombre est moindre. Quant aux règles, elles doivent prendre en compte les groupes de caractères ("ch", "ein", "ou" en français) et les

.../...

variations contextuelles ("g" dans "gave" et "genre"). On peut aussi vouloir prononcer les nombres écrits en chiffres arabes. Et en français il faut assurer les liaisons car certaines sont absolument obligatoires. Enfin, pour une synthèse plus naturelle on doit aussi déterminer la prosodie, c'est-à-dire le rythme, les accents d'intensité et l'intonation.

On s'aperçoit alors que dans un certain nombre de cas il faudrait une analyse syntaxique pour résoudre des ambiguïtés ("le couvent" mais "elles couvent", "des rations" mais "nous rations"). Les liaisons et la prosodie dépendent aussi de la structure syntaxique. Il y a même des cas où il faudrait également une analyse sémantique ("les fils" pluriel de "fil" ou de "fils" ?) . Malheureusement, une analyse syntaxique même sommaire exige des algorithmes compliqués avec un temps de calcul important. Et une analyse sémantique serait encore plus complexe.

Or on remarque que l'amélioration apportée est faible statistiquement par rapport à la lourdeur du traitement supplémentaire à effectuer. En fait, sans analyse syntaxique ni sémantique on peut atteindre plus de 95% de bons résultats. C'est surtout vrai si on utilise des procédés heuristiques comme le choix le plus probable, le découpage par la ponctuation ou les mots-outils, la longueur des mots, etc... Même s'il est théoriquement impossible d'atteindre ainsi un résultat parfait, en pratique on peut raisonnablement s'en contenter pour une conversion rapide.

Effectivement la plupart des programmes de traduction phonétique utilisés ne font pas d'analyse syntaxique, ou se limitent à un découpage rudimentaire, de façon à obtenir un résultat en temps réel. C'est le cas au LIMSI du programme de D. Teil, le premier réalisé en France. Ce programme peut s'utiliser avec d'autres synthétiseurs, mais il a été incorporé au synthétiseur à canaux "Icophone V", qui est entièrement autonome car il comprend un mini-ordinateur pour sa commande avec les fichiers de paramètres et le programme de traduction en mémoires mortes.

Cet appareil de format réduit peut donc prononcer lui-même un texte écrit. Le programme a été ensuite amélioré, en particulier pour introduire la prosodie. De même, on peut mentionner les programmes de traduction réalisés au CNET à Lannion et à l'Ecole des Télécommunications à Paris. Par contre, les programmes élaborés par J. Vaissière pour le français ou par J. Allen pour l'anglais sont plus complexes mais aussi plus lents.

.../...

référence
manuscrite
dans la
bibliothèque

3) RECONNAISSANCE

L'analyse et la synthèse ont permis d'établir avec précision la structure acoustique de la parole. Grâce aux connaissances ainsi acquises, on put alors dans les années 50 aborder le problème de la reconnaissance de la parole, c'est-à-dire tenter d'identifier automatiquement les phonèmes, les mots, les phrases de la langue parlée. C'est un problème difficile mais par là même très intéressant, car on doit simuler non seulement la perception acoustique mais aussi les niveaux supérieurs linguistiques et sémantiques qui conditionnent la perception.

C'est un excellent exemple des difficultés souvent inattendues que pose l'automatisation d'une fonction mentale, et des connaissances nouvelles que cette démarche apporte. On est amené à utiliser les acquis de la linguistique mais en même temps à découvrir des mécanismes psycholinguistiques d'une complexité insoupçonnée que seules les exigences de la simulation obligent à mettre en évidence. On doit d'ailleurs admirer les performances et les capacités d'adaptation de ces mécanismes humains complexes, qui pourtant nous paraissent tout naturels alors qu'il est si ardu de les simuler de façon même très limitée. Bien entendu, c'est l'ordinateur qui permet ce genre de recherches en pratique.

De même que l'oreille humaine effectue une analyse acoustique préalablement à toute identification, la première étape de la reconnaissance automatique consiste en une analyse physique du signal de parole. On utilise donc une des méthodes qu'on a vues au chapitre sur l'analyse : filtres analogiques ou numériques, transformée de Fourier rapide (FFT), codage prédictif, etc... On obtient ainsi une série de paramètres simples se déroulant au cours du temps. Mais ce pré-traitement indispensable reste purement physique et ne résoud en rien le problème essentiel de l'identification des éléments pertinents de la structure proprement linguistique du message.

Car ce qu'il faut reconnaître, ce sont les éléments signifiants du code parlé (traits distinctifs, phonèmes, mots, prosodie...) et non pas l'organisation physique du signal acoustique. Or la correspondance entre la forme sonore et les unités linguistiques abstraites est complexe, car les formes varient selon le contexte et les locuteurs, et l'identification est loin d'être évidente.

3-1 Identification phonétique

. Reconnaissance globale et analytique

En fait il y a deux approches différentes selon le niveau où on se place et les éléments qu'on cherche à identifier : la reconnaissance globale par mots, et la reconnaissance analytique. Dans la reconnaissance globale on reconnaît des expressions ou des mots isolés sans les décomposer phonétiquement. On ne peut donc identifier que des mots enregistrés à l'avance et le vocabulaire est forcément limité (une centaine de mots en général). Mais reconnaître des unités plus longues comme les mots est relativement plus facile, et cette méthode est maintenant opérationnelle pour un seul locuteur (celui qui a enregistré le vocabulaire).

Par contre la reconnaissance analytique, où on cherche à identifier les unités minimales phonétiques (phonèmes et traits distinctifs), s'est révélée plus difficile, alors que théoriquement elle permettrait de reconnaître n'importe quelle phrase sans limitation de vocabulaire. On a beaucoup de mal pour segmenter le flot continu de la parole en éléments phonémiques, puis pour identifier ces éléments qui varient énormément avec le contexte sonore, le locuteur, le débit... On peut comparer au problème de la reconnaissance automatique de l'écriture manuscrite, qui est également encore mal résolu pour des raisons semblables de variabilité des éléments, mais qu'on peut se représenter peut-être plus clairement. En tous cas la reconnaissance analytique de parole continue est loin d'être maîtrisée à l'heure actuelle.

. Variabilité de la parole

En effet la variabilité de la parole est très grande, selon le sexe et l'âge du locuteur, son timbre de voix et son accent régional, et pour un même locuteur selon la vitesse d'élocution, la netteté d'articulation, le niveau sonore, etc... Par exemple la hauteur mélodique et l'échelle des formants deviennent nettement plus aigus quand on passe d'une voix d'homme à une voix de femme puis à une voix d'enfant. La durée d'un mot dépend du débit, la netteté diminue avec la fatigue. Voix chuchotée et voix criée diffèrent aussi beaucoup du registre normal. Et pourtant un être humain n'est en général pas très gêné pour comprendre par toutes ces différences, même s'il les perçoit et les identifie comme des caractéristiques personnelles ou temporaires du locuteur.

De plus, même pour un locuteur unique, les unités phonémiques varient suivant le contexte. Du point de vue articulatoire et acoustique les sons de la parole sont liés en un flot continu et s'influencent les uns les autres. En particulier la plupart des consonnes se réalisent plus comme des modifications des voyelles adjacentes que comme des phénomènes indépendants. En fait la perception d'unités discrètes ne se fait que grâce à des opérations psychologiques complexes de segmentation, de comparaison et de classification des formes sonores.

Comme l'ont fait remarquer les phonologues structuralistes, les unités linguistiques (phonèmes et traits distinctifs) sont des éléments fonctionnels abstraits, et il n'est pas facile de définir leur réalisation physique. C'est pourquoi la reconnaissance globale est relativement plus facile que la reconnaissance analytique, car on évite le problème des variations contextuelles. Mais on reste gêné par le changement de locuteur, puisque les voix diffèrent.

On a donc mis au point des algorithmes astucieux qui tiennent compte de toutes ces variations. Après une analyse acoustique on cherche à extraire des paramètres pertinents comme la position des formants, les transitions, le voisement, l'intonation, afin de réduire la quantité d'information à traiter. Puis on segmente cette séquence de paramètres et on effectue des normalisations temporelles et fréquentielles avant de tenter de classer les segments obtenus.

On utilise des méthodes diverses mais en général on calcule des taux de comparaison complexes afin d'identifier la place relative d'un élément parmi les choix possibles. En effet la variabilité de la parole rend illusoire d'arriver à une identification absolue, et il a fallu élaborer des programmes de plus en plus compliqués, à la mesure de la complexité des mécanismes perceptifs que l'on veut ainsi reproduire.

3-2 Niveaux supérieurs

. Structure linguistique

Et pourtant, malgré tous ces efforts il semble bien qu'on se heurte à une limite. En pratique on n'arrivait pas à dépasser un taux de reconnaissance de 70 à 80 % en parole continue, ce qui est insuffisant. En fait c'est qu'on avait négligé un problème fondamental en considérant la parole comme une simple suite phonétique et en oubliant les règles du langage humain.

Car le langage est fortement structuré en une hiérarchie de composants emboîtés, avec à chaque niveau des règles qui réduisent de beaucoup les possibilités de combinaisons. Les traits se regroupent en phonèmes, les phonèmes en mots, les mots en phrases, les phrases en discours, pour ne citer que les divisions les plus grossières, et ces groupements obéissent toujours à des règles d'association. Ces restrictions combinatoires constituent évidemment une aide précieuse à la compréhension et doivent donc être intégrés à un système de reconnaissance automatique.

Certes un locuteur humain est théoriquement capable de prendre une dictée phonétique de suite de syllabes sans signification aucune, mais seulement dans des conditions idéales de prononciation et d'attention qui ne sont pas vérifiées dans une conversation normale. En revenant à la comparaison avec l'écriture manuscrite, on peut remarquer qu'il est plus difficile d'identifier les caractères (d'un alphabet connu) d'un texte écrit à la main dans une langue qu'on ne connaît pas, parce qu'on est alors privé des contraintes d'ordre supérieur, alors qu'on lira sans mal un texte écrit de la même main dans une langue connue.

Car sachant par expérience qu'il sera compris, l'auteur de la lettre a tout naturellement déformé par économie d'effort les distinctions graphiques. De même le locuteur sait bien qu'il n'est pas nécessaire de prononcer parfaitement pour être compris et relâche son articulation autant que possible, d'autant plus qu'en dialogue le contexte de situation et les réactions de son interlocuteur lui permettent de doser son effort. On ne peut donc pas identifier dans la chaîne sonore des distinctions phonétiques qui ne s'y trouvent pas, d'où la limitation du taux de reconnaissance en parole normale. Cela paraît une limite d'ordre fondamental, même s'il faut bien sûr essayer d'améliorer encore la reconnaissance proprement phonétique.

On doit donc simuler non seulement la perception phonétique mais aussi les niveaux supérieurs d'organisation linguistique des messages. Plus précisément, il a fallu tenir compte de ce que la perception phonétique elle-même est conditionnée par les contraintes dues aux niveaux supérieurs. C'est vers les années 60 qu'on s'en est aperçu après les illusions simplistes du début des recherches, et on a alors commencé à introduire ces contraintes dans les programmes de reconnaissance.

.../...

On ne cherche plus à reconnaître une chaîne de phonèmes dès le niveau phonétique. On se contente de garder plusieurs candidats possibles avec des notes de vraisemblance, pour chaque segment découpé dans la chaîne sonore. On repousse donc la décision finale aux niveaux supérieurs, qui permettront de choisir entre les interprétations possibles en cheminant dans le réseau des candidats provisoires. Le processus devient plus compliqué mais aussi plus souple et plus performant.

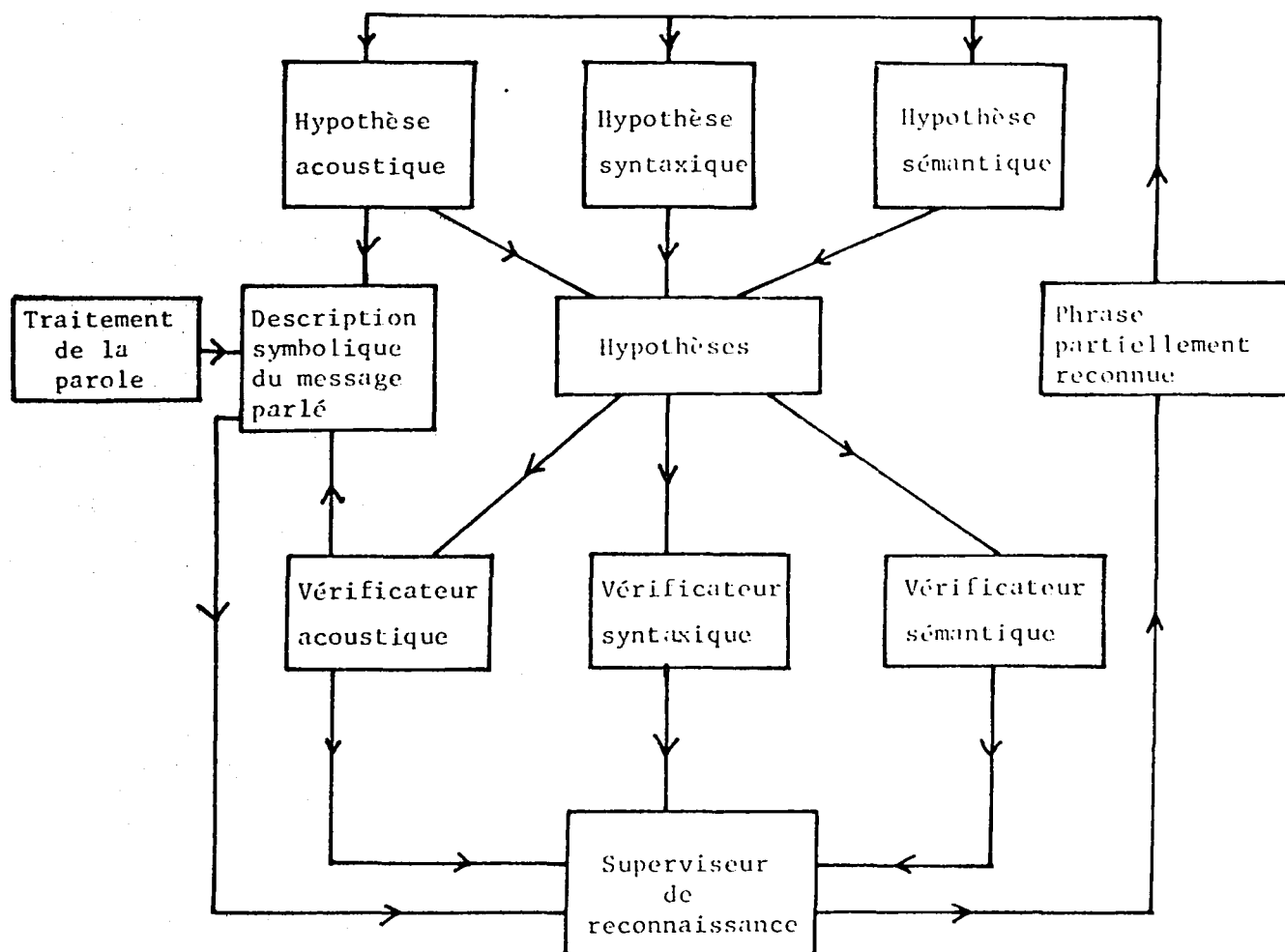
Ainsi on peut utiliser des règles phonologiques. Par exemple en français un mot ne peut pas commencer par les groupes /tl/ ou /bk/, ou en anglais un mot ne peut pas se terminer par une voyelle relâchée accentuée. Par contre il faut prévoir des phénomènes comme l'assimilation ou l'élision de phonèmes adjacents. Il y a des contraintes lexicales : à un moment donné de la langue il n'existe dans le lexique qu'une partie seulement des mots théoriquement permis par les règles phonologiques. Les mots "vilure" et "boutaque" n'existent pas en français actuel et doivent être éliminés si on croyait les reconnaître.

Puis les règles de la syntaxe limitent les combinaisons de mots possibles : "J'ai mal au pied" est plus vraisemblable que "J'ai molle au piller" ou "Chez mal loup pied". Enfin il y a des contraintes sémantiques et pragmatiques, mais qui semblent plus difficiles à généraliser car elles dépendent du contexte et des connaissances qu'on peut avoir. Par exemple dans une application spécifique au jeu d'échec, il est peu probable qu'on demande à déplacer les pièces hors des limites de l'échiquier ou contrairement aux règles du jeu. Ceci dit les processus de décision doivent rester assez souples pour tolérer une mauvaise prononciation ou des constructions incorrectes de la part du locuteur, ainsi que des erreurs de segmentation ou d'identification au niveau phonétique. Il faut aussi pouvoir mettre à jour le contexte sémantique au cours du dialogue.

. Hiérarchie des niveaux

C'est ainsi que s'est posé le problème de l'organisation de ces différents niveaux. On distingue des systèmes hiérarchiques effectuant une analyse montante ou descendante, et d'autre part des systèmes parallèles et interactifs. Les systèmes hiérarchiques supposent que les niveaux phonétique, lexical, syntaxique et sémantique sont organisés selon un ordre strict sans possibilités de rétroaction. L'analyse montante ("bottom-up") essaye de tirer des mots du réseau phonétique, puis de regrouper les mots possibles

.../...



Le système de reconnaissance HEARSAY
utilisant le paradigme hypothèse-vérification sur des niveaux non hiérarchisés

en phrases possibles et de vérifier la validité sémantique.

Malheureusement, à moins d'une excellente reconnaissance phonétique et de contraintes lexicales et syntaxiques très strictes, les possibilités se multiplient à chaque niveau successifs de façon exponentielle, et on assiste à une explosion combinatoire. Le temps de calcul devient vite prohibitif sinon impossible. Quant à l'analyse descendante ("top-down") elle part d'hypothèses aux niveaux supérieurs pour essayer de les vérifier aux niveaux lexical et phonétique. Mais cela suppose un contexte très contraignant et une syntaxe limitée, sinon on ne peut pas proposer d'hypothèses. Les systèmes hiérarchiques ne paraissent donc vraiment applicables que dans des cas assez particuliers.

On s'oriente alors de plus en plus actuellement vers des systèmes parallèles où les niveaux ne sont pas hiérarchisés mais en interaction constante. On considère que chaque niveau doit pouvoir utiliser à la fois les informations venant des niveaux inférieurs et les prévisions ou contraintes des niveaux supérieurs. C'est-à-dire qu'on emploie à la fois l'analyse montante et l'analyse descendante de manière à limiter les possibilités et éviter l'explosion combinatoire. Chaque niveau peut intervenir à tout moment selon les besoins pour émettre et vérifier des hypothèses diverses, et la rétroaction est généralisée.

Ainsi la perception phonétique est conditionnée par les hypothèses que font les niveaux supérieurs en fonction du contexte, et inversement ces prévisions sont corrigées d'après la perception phonétique. Certes, ce modèle parallèle pose des problèmes de synchronisation entre les niveaux, mais il paraît plus efficace et sans doute plus proche de la perception humaine.

Ceci dit, ces distinctions sont plutôt théoriques, car les systèmes réellement mis au point combinent souvent plusieurs approches, et il n'est pas possible de les classer sans un certain arbitraire. Ainsi, pour ne citer que des équipes ayant participé au projet ARPA dans les années 70, les systèmes "HEARSAY" réalisés à Carnegie Mellon University sont un bon exemple d'organisation parallèle, mais le système "SPEECHLIS" de Bolt Beranek & Newman est en partie ascendant et en partie descendant.

Enfin, il faut répéter que la reconnaissance automatique n'est pas un problème résolu à l'heure actuelle. Les systèmes ne fonctionnent que pour un seul locuteur avec un vocabulaire limité, une syntaxe simple et un contexte spécifique. Ce sont là des contraintes sévères. Mais les

recherches sont en pleine évolution, dans les deux directions complémentaires de la reconnaissance phonétique et de l'intervention des niveaux supérieurs du langage. On peut donc raisonnablement prévoir des progrès importants dans un avenir proche.

III - GENERATION DES PHRASES

1) INTRODUCTION

Dans les chapitres précédents on a tenté^{de}/donner une vue d'ensemble du traitement automatique de la parole en général. Mais dans les chapitres suivants on va décrire surtout le programme de génération de phrases qui constitue l'essentiel de cette étude. Après avoir exposé les choix de base, on détaillera la structure et le fonctionnement du programme. En même temps on essaiera de généraliser les problèmes soulevés et d'en tirer des conclusions utiles pour l'avenir.

1-1 Génération automatique

Le dialogue par la parole entre l'homme et l'ordinateur demande l'exécution coordonnée d'un nombre de tâches très diverses. On a réparti plus ou moins arbitrairement ces tâches de la chaîne de parole en différents modules. Parmi ces modules, on vient de voir en détail l'analyse, la synthèse, et la reconnaissance automatique de la parole. Mais si la chaîne d'analyse et de reconnaissance permet bien à la machine de recevoir des messages parlés, il reste à en extraire le sens puis à engendrer la réponse. Or la synthèse acoustique ne peut que prononcer des phrases données au préalable en transcription phonétique (ou en orthographe ordinaire).

Pour que la machine puisse vraiment émettre elle-même des messages non prévus à l'avance, il faut qu'elle soit aussi capable d'engendrer les phrases à prononcer exprimant le sens requis par le dialogue en cours. En résumé, il faut pouvoir passer du son au sens, puis du sens au son tout comme l'interlocuteur humain. Mais il y a là deux problèmes différents bien qu'étroitement liés : la détermination du sens d'une part, puis la génération des messages correspondants. Le premier problème est celui de la sémantique proprement dite, et on sait que c'est un sujet difficile, mal connu et extrêmement controversé.

.../...

En effet, il n'y a pas actuellement en sémantique de consensus même partiel sur les questions fondamentales. On n'est d'ailleurs pas sûr des limites de la sémantique par rapport à la syntaxe et à la pragmatique, et on peut même se demander si la sémantique se distingue du raisonnement en général. Malgré des discussions sans fin en philosophie, psychologie, linguistique, logique et informatique, il ne semble pas se dégager d'accord. On oscille entre des thèses générales plutôt vagues, et des traitements très spécifiques d'un problème particulier. Il faut bien avouer que la vieille question des rapports entre le langage, la logique, et la pensée n'a pas encore reçu de réponse vraiment satisfaisante. En conséquence le choix des éléments de sens et de leur représentation reste largement discutable.

On ne prétend pas dans ce travail apporter du nouveau sur cette question. On considère comme résolu le problème de ^{comment} déterminer des énoncés sémantiques à partir de l'état interne de la machine au cours du dialogue, en fonction des capacités déductives du système, de ses connaissances, des questions posées, de la tâche commandée, etc... On a alors choisi une représentation sémantique qui puisse être engendrée par un système déductif, c'est-à-dire une formulation de type logique. Mais on se contente de considérer les énoncés sémantiques dans cette formulation comme une donnée pour le programme de génération, et on n'essaye pas de les obtenir automatiquement par ailleurs.

On a donc ainsi décomposé le problème de génération en deux parties successives pour ne s'attaquer d'abord qu'à l'étape postérieure. Il peut paraître bizarre de commencer par la fin, mais quand on étudie le langage naturel on est bien obligé de partir des phénomènes de surface, seuls visibles, pour tenter d'aller vers le sens profond, qui est inobservable. Certes, en se rapprochant ensuite des processus sémantiques on pourra être amené à revoir la formulation sémantique. Mais on a bien envisagé la possibilité d'un développement ultérieur en choisissant la formulation. On peut espérer assurer la compatibilité dans ce cas par une traduction formelle simple et rapide.

Par contre la génération des phrases phonétiques à partir des énoncés sémantiques a été traitée aussi complètement que possible, et représente l'essentiel de ce travail. On peut voir le problème comme une traduction à effectuer entre le langage des formules sémantiques et la langue naturelle parlée. Cela peut se comparer aux manipulations formelles nécessaires pour la traduction automatique des

langues naturelles, la compilation des langages de programmation, et les traductions de code en général. On a besoin des règles d'interprétation du langage source, d'un dictionnaire de référence, et des règles de génération du langage but. C'est bien le contenu du programme de génération.

1-2 Programme de génération

La génération de phrases est alors réalisée par un programme écrit en langage PL/1, et implémenté sur ordinateur IBM 370/168 en système MVS. Mais le programme ne fait pas appel aux particularités du système et devrait donc se révéler portable.

Le langage utilisé a été choisi surtout pour des raisons pratiques : PL/1 est un langage évolué très répandu qui dispose de compilateurs performants et fiables. C'est un outil puissant qui permet en particulier de traiter facilement les chaînes de caractères et les listes. Il est donc bien adapté aux traitements linguistiques, et on l'a préféré à un langage plus spécifique de traitement de listes comme LISP qui est moins maniable, car non standardisé et peu optimisé.

Le programme de génération accepte en entrée des énoncés sémantiques exprimant le sens des messages à prononcer. Les énoncés sont donnés dans une formulation fonctionnelle dont on verra la notation exacte. Le choix de cette formulation mérite d'ailleurs discussion, car il peut paraître assez arbitraire. Ceci dit, le programme lit ces énoncés et en tire une représentation interne du sens sous forme d'arborescence en mémoire.

Mais pour que le programme puisse traiter les énoncés sémantiques, il faut avoir construit auparavant un dictionnaire à partir d'entrées lexicales. Ces entrées indiqueront pour chaque notion sémantique les caractéristiques grammaticales et la prononciation du mot français correspondant. Mais si on doit fournir ces données lexicales au programme, la construction du dictionnaire est alors automatique et celui-ci est facile à modifier.

En sortie le programme génère une phrase en français exprimant le sens de la formule sémantique donnée en entrée. Le résultat est en langue parlée : il n'est pas fourni en orthographe écrite, mais en notation phonétique. On génère ainsi toute l'information nécessaire (phonèmes et marqueurs prosodiques) à la prononciation de la phrase par un synthétiseur de parole, sans passer par le texte écrit.

La sortie n'est donc pas autre chose qu'une chaîne de symboles phonétiques, dans un code utilisant les caractères disponibles sur l'équipement informatique standard. Ce code, où chaque phonème est représenté par un seul caractère, est celui recommandé par l'ensemble des équipes de recherche sur la parole en France.

Pour obtenir la phrase phonétique à sortir, le programme parcourt d'abord l'arborescence sémantique construite à la lecture de l'énoncé donné en entrée. Ce faisant il consulte le dictionnaire pour trouver les renseignements indispensables à la génération de la phrase parlée. Cette première passe permet de déterminer les genres, nombres, et personnes nécessaires pour les accords. Puis dans une deuxième passe on effectue tous les accords morphologiques et on place les mots accordés selon les règles de la syntaxe. Enfin, on traite les phénomènes de surface comme liaisons, élisions, contractions, qui ne dépendent que de l'ordre final des mots dans la phrase. Il ne reste plus alors qu'à transmettre le résultat phonétique au synthétiseur.

Cette partie morpho-syntaxique du programme, qui travaille à partir de l'arborescence sémantique et du dictionnaire, constitue la partie la plus importante. En effet elle représente une grammaire du français oral. Bien sûr ce n'est en fait qu'un petit sous-ensemble de la grammaire naturelle, et le programme n'engendre que des phrases simples. Mais on sait qu'une grammaire exhaustive devrait comporter un nombre énorme de règles, et c'est un travail qui n'a encore jamais été vraiment mené à bien pour aucune langue. On s'est donc limité en choisissant une morphologie et une syntaxe de base, suffisantes pour produire des messages simples mais naturels.

Ensuite il faut souligner que c'est une grammaire orale. Toutes les règles morphologiques et syntaxiques, ainsi que le dictionnaire, concernent uniquement la forme orale de la langue. Même s'il y a un certain parallélisme avec le français écrit, les règles orales diffèrent des règles écrites, et ne sont pas données par les grammaires. Car par tradition scolaire grammairiens et linguistes ont surtout étudié la langue écrite, alors que pour des raisons historiques la forme orthographique du français est assez éloignée de la prononciation. Avant toute réalisation informatique, il a donc fallu faire une recherche linguistique originale pour établir les règles orales.

En particulier, la morphologie du français oral est à la fois complexe et nettement différente du français écrit. Le fait que le français est une langue à morphologie riche impose alors un travail important. Il faut aussi résoudre le problème des liaisons, à peine visible dans la forme écrite, mais pourtant caractéristique du français parlé. De même la prosodie n'est que mal suggérée par les signes de ponctuation, alors qu'elle est essentielle à la prononciation. Et on pourrait citer bien d'autres exemples ...

C'est cet aspect purement oral de la grammaire utilisée qui fait l'originalité de ce programme sur le plan linguistique. C'est une différence essentielle par rapport à la génération de phrases dans des programmes comme "Eliza", ou celui de Winograd, les systèmes d'interrogation de fichiers ou même la traduction automatique. Mais cela semble la façon la plus évidente de traiter le problème de génération pour un dialogue en langue parlée. C'est d'ailleurs ce qui permet de résoudre de manière élégante des problèmes comme les liaisons ou la prosodie, qui exigeraient autrement une lourde analyse syntaxique sur le texte écrit. Enfin ce traitement oral est sans doute plus proche de ce qui se passe chez le locuteur humain lors de la production de parole.

2) STRUCTURES DE LISTES

Les listes sont un outil essentiel en programmation non-numérique, et en particulier dans les problèmes d'intelligence artificielle où on traite des expressions formelles ou du langage naturel. Le programme de génération utilise largement les listes, et il est donc indispensable d'explicitier la conception de ces structures de données avant de poursuivre l'exposé.

2-1 Organisation des données

Les structures de données sont des modes d'organisation des données utilisées, permettant à la fois de les traiter en bloc et d'avoir accès à une donnée particulière. Les structures les plus usuelles sont les tableaux indicés, et aussi dans certains langages les structures hiérarchisées. Dans ces structures l'accès aux données est très facile, mais en modifier l'organisation est peu commode. Par exemple pour insérer

un nouvel élément dans un tableau on est obligé de modifier toute la suite du tableau. C'est une organisation trop rigide pour manipuler aisément des expressions de forme variable.

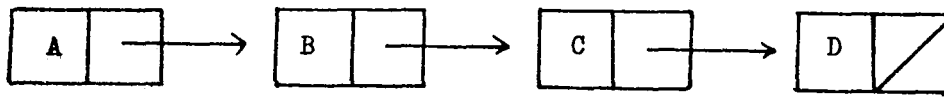
Par contre les structures de listes offrent une grande souplesse. Il est facile de les créer, de les étendre, d'y insérer ou d'en extraire un élément, etc... Ceci est dû à ce que l'adresse de chaque élément d'une liste est donnée explicitement par l'élément précédent grâce à un pointeur associé. Plus précisément, tout élément d'une liste est un doublet unissant une information à un pointeur contenant l'adresse de l'élément suivant. Par convention on représente ce lien par une flèche allant du pointeur au doublet suivant. Ce chainage par adresses est indépendant de la place réelle des informations dans la mémoire, car il ne dépend que du contenu des pointeurs de la liste. Ainsi il est facile de modifier la succession des éléments d'une liste en modifiant les adresses dans les pointeurs.

On voit la puissance et la souplesse d'une telle structure somme toute assez simple, pour organiser des données ou représenter un graphe. Mais il faut bien sûr savoir gérer les listes pour pouvoir s'en servir utilement, ce qui pose de nombreux problèmes techniques. Par exemple il faut une tête de liste pour repérer la liste, et une valeur particulière de pointeur pour marquer la fin d'une liste. Mais sans entrer trop dans les détails, on se contentera d'exposer quelques opérations élémentaires nécessaires pour manier les listes.

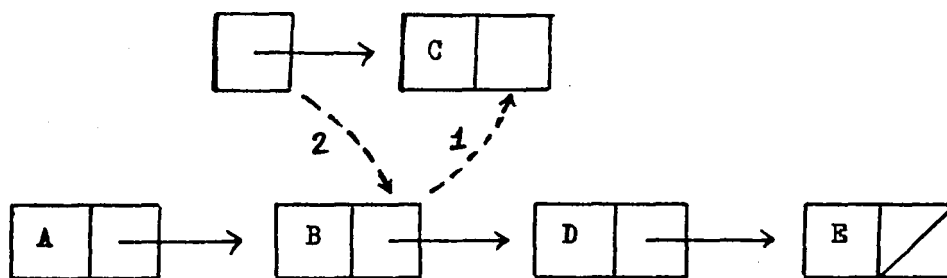
2-2 Opérations élémentaires

Le choix des opérations primitives de traitement de listes est assez arbitraire. Il dépend du langage utilisé, qui peut offrir des facilités pré-existantes plus ou moins élaborées (pointeurs, allocation dynamique, etc...), et des besoins particuliers du programme. Il y a un compromis à trouver entre la simplicité d'écriture permise par des primitives plus puissantes, et la difficulté de définir des actions adaptées à des besoins qui peuvent varier. Mais en tous cas il faut au moins pouvoir allouer des éléments, créer une liste, l'étendre, la modifier par insertion ou extraction, la parcourir.

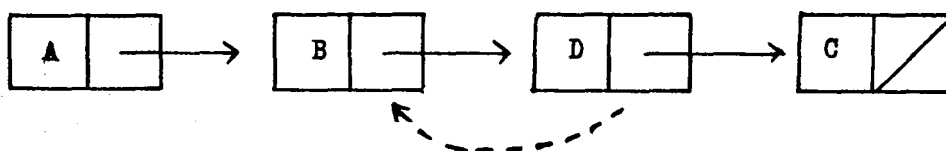
Pour créer une liste, il faut allouer les éléments composants et les chaîner en faisant pointer chaque élément sur le suivant. On étend la liste en ajoutant des éléments de la même manière.



Structure de liste



Insertion d'un élément



Extraction d'un élément

Pour insérer un nouvel élément, on le fait pointer sur l'élément suivant (en recopiant le contenu du pointeur précédent après lequel on insère) puis on recopie l'adresse du nouvel élément dans le pointeur précédent (voir la figure). Pour extraire un élément, il suffit de l'isoler du chainage en recopiant son pointeur dans le pointeur de l'élément précédent. Enfin pour parcourir la liste on suit le chainage de pointeur en pointeur. Toutes ces actions ne jouent que sur les adresses et sont assez simples.

En fait on a surtout utilisé les actions fournies par le langage de programmation choisi. En effet, le langage PL/1 permet de gérer automatiquement l'allocation de variables pointées et leur adressage. Les opérations élémentaires de listes en découlent aisément dès qu'on sait manier les dispositions du langage lui-même. Pour éviter des choix précoces, on n'a d'abord élaboré que des primitives très simples de parcours et d'insertion. Mais on pourrait maintenant sans doute améliorer le programme en mettant au point des primitives plus complexes adaptées aux problèmes à traiter qui bien souvent ^{en}se révèlent qu'à l'usage.

3) REPRESENTATION SEMANTIQUE

Pour générer par ordinateur des phrases exprimant un sens donné, il faut d'abord disposer d'une représentation formelle du sens. Car il semble bien que la formalisation soit nécessaire pour le traitement automatique. Or c'est justement par le langage naturel que nous exprimons généralement des significations. Mais les langues naturelles sont imprécises, ambiguës, arbitraires, complexes et irrégulières, et on pense depuis longtemps qu'elles ne sont sans doute que l'expression de surface d'un système sous-jacent plus rigoureux et plus facile à formaliser. Malheureusement il n'y a pas d'accord général sur la forme exacte d'un tel système, qui demande donc une discussion préalable.

3-1 Choix d'une représentation

On peut supposer qu'une représentation du sens différente du langage ordinaire correspond bien à une réalité psychologique, mais ce sens profond n'est pas directement observable, et reste du domaine des hypothèses. D'où la diversité des propositions, d'Aristote à Chomsky,

sur le système profond qui serait sous-jacent à l'utilisation du langage. Il n'est d'ailleurs pas certain qu'on parle de la même chose d'une discipline à l'autre ! Philosophes et logiciens ont tendance à aller beaucoup plus loin dans l'abstraction. Et en fait la logique symbolique moderne cherche à formaliser le raisonnement mathématique plutôt que le langage ordinaire, même si cette forme est souvent abstraite d'éléments du langage naturel. L'application de la logique à la description linguistique est donc loin d'être convaincante, malgré de nombreuses tentatives souvent intéressantes.

Par contre les linguistes, plus conscients de la complexité du langage, sont en général nettement plus prudents. Certains refusent même de s'occuper de sémantique par principe. Et même dans la grammaire transformationnelle de Chomsky, les structures profondes restent malgré leur arbitraire très proches des structures de surface. Ce qui a conduit les partisans de la sémantique générative à proposer des structures sémantiques beaucoup plus abstraites. Mais en dépit de nombreuses discussions, l'apport des linguistes à la sémantique n'est pas très satisfaisant. On peut alors espérer que des travaux en intelligence artificielle comme ceux de Winograd apporteront davantage par leur approche empirique et opératoire que des discussions trop théoriques....

En bref, on n'a pas encore vraiment résolu les grandes questions que pose la sémantique. Quel est le sens exprimé par le langage ? Comment peut-on le représenter ? Peut-on le formaliser ? Par la logique ? Et quel serait le système logique le plus adéquat ? Quels sont les éléments de base ? Et la pensée est-elle dépendante du langage ? On n'a pas de réponse sûre pour aucune de ces questions pourtant essentielles. On ne sait pas non plus si on peut généraliser les résultats partiels obtenus par des études particulières. On hésite donc toujours sur le choix des éléments de sens et sur leur représentation. X

Il est d'ailleurs possible que ces questions soient restées insolubles, comme beaucoup de problèmes philosophiques, non seulement à cause du manque de connaissances pertinentes mais aussi parce qu'elles sont tout simplement mal posées. Ainsi on peut se demander s'il existe réellement une sémantique linguistique proprement dite, et si la compréhension du langage est fondamentalement différente de l'interprétation du monde en général. S'il n'y avait pas de système sémantique

.../...

particulier au langage, cela expliquerait qu'on aie tellement de mal à en donner une description précise et cohérente. En tous cas il semble bien que l'interprétation sémantique d'énoncés linguistiques fasse nécessairement appel aux capacités déductives et aux connaissances générales du locuteur. La représentation du sens d'un énoncé serait alors inséparable d'un vaste réseau de connaissances pragmatiques et donc difficile à formaliser de manière complète.

En effet le langage naturel est un moyen de communication entre êtres humains, qui disposent d'une intelligence déductive et d'un grand nombre de connaissances pratiques ou culturelles. Il n'est donc pas nécessaire de spécifier explicitement ce qui est connu ou prévisible pour un esprit humain, d'où l'apparente imprécision des langues naturelles. Le langage humain n'est pas un système formel justement parce qu'il s'adresse à des êtres humains, et non à des machines à qui il faut tout expliquer en détail. C'est ce qui explique les échecs des premières tentatives de traitement informatique du langage comme la traduction automatique. L'utilisation du langage est inévitablement liée à des raisonnements extra-linguistiques, et il est très probablement futile de tenter d'assigner une limite précise entre pragmatique et sémantique, ou même entre sémantique et syntaxe.

En conclusion, le choix d'une représentation sémantique ne peut qu'être assez arbitraire étant donné l'état actuel du problème. Il faut le reconnaître, mais on peut alors adopter une attitude empirique. Sans perdre de vue les questions théoriques, on choisira une représentation en fonction des besoins pratiques de l'application envisagée. La représentation choisie aura ainsi un caractère expérimental et c'est à l'usage qu'on pourra juger de sa validité, essayer de l'améliorer, et peut-être de la généraliser en l'étendant à d'autres domaines. L'intérêt de la modélisation sur ordinateur est justement de permettre de tester sérieusement les conséquences d'un tel choix, et de vérifier ainsi des hypothèses théoriques.

Il faut donc voir quelle sorte de représentation conviendrait comme donnée du programme de génération. Pour que ce programme puisse interpréter facilement les énoncés sémantiques, une formalisation rigoureuse serait souhaitable. On doit aussi se souvenir que ces énoncés sont supposés provenir d'un système déductif. On a laissé de côté le problème de la production des énoncés sémantiques, mais les systèmes déductifs travaillent généralement sur des formules de type

.../...

.../...

logique. Il faudrait donc une formulation proche de la logique, qui soit connue et assez générale pour ne pas se figer. Enfin cette forme logique doit permettre la génération de langage naturel en ayant un contenu linguistique. En particulier il faut faire les distinctions exigées par la grammaire de la langue d'arrivée comme par exemple le nombre ou la personne en français.

3-2 Notation fonctionnelle

Ces diverses considérations nous ont amené à choisir pour la représentation sémantique une formulation familière, la notation fonctionnelle. Toute relation est représentée comme une fonction ayant un certain nombre d'arguments : $f(x_1, x_2 \dots x_n)$, et une fonction peut elle-même servir d'argument à une autre fonction :

$g(f(x))$. Cette notation permet de formuler assez simplement les relations grammaticales les plus variées, et comme elle est bien connue en logique, on bénéficie des études déjà effectuées. En particulier on peut se rattacher au travail considérable accompli en logique des prédicats.

Cette notation étant parenthésée, il n'est pas nécessaire de connaître à l'avance le nombre d'arguments que prend un prédicat. Ceci facilite la lecture et permet d'ajouter des prédicats nouveaux à tout moment, mais il faut bien observer le parenthésage. C'est un inconvénient par rapport à la notation polonaise par exemple, qui est moins lisible mais plus compacte. Ainsi en première approximation, on représentera le sens de la phrase simple "Le jeune enfant aime la soupe" par la formule :

AIMER (JEUNE (ENFANT), SOUPE)

Il faut remarquer que les éléments de la formule sont des notions sémantiques considérées comme élémentaires, et non pas des mots du français. L'élément "ENFANT" représente le concept d'enfant et non le mot français "enfant". C'est par commodité qu'on utilise la même orthographe, et aussi bien sûr parce qu'il est difficile de distinguer entre les notions sémantiques et les mots d'une langue donnée.

.../...

On retrouve là le vieux problème philosophique des rapports entre la pensée et le langage, et cette ambiguïté est pratiquement inévitable dans de nombreux cas.

Mais d'autre part on utilisera comme éléments sémantiques des concepts auxquels ne correspondent pas de mots lexicaux car ils sont exprimés en français par des moyens grammaticaux comme le pluriel, l'interrogation... Inversement on peut estimer que des mots comme l'auxiliaire "être" ou les prépositions "à" ou "de" sont le plus souvent vides de sens et ne doivent pas figurer dans les énoncés sémantiques, quitte à être introduits par la suite dans la phrase à générer. Les éléments sémantiques ne coïncident donc pas vraiment avec le lexique du français.

Par exemple le prédicat NEG indiquera un énoncé négatif, QU signalera une question, PLU le pluriel. L'article étant nécessaire en français, il faut aussi indiquer par DEF si un nom est défini ("le, la, les"), par IND s'il est indéfini ("un, une, des"), par PART s'il est partitif ("du, de la, des"). On sait d'ailleurs que la différence entre notions lexicales et notions grammaticales n'est pas nette : l'article est-il un mot ou bien une catégorie ? En tous cas il faut faire dans les énoncés sémantiques les distinctions qui sont obligatoires dans la langue choisie. Ainsi il faut préciser en français l'article pour le nom, ce qui est inutile en russe, et la personne pour le verbe, ce qui n'est pas nécessaire en chinois ou en japonais.

Donc la phrase simple vue plus haut "Le jeune enfant aime la soupe" devra en fait être exprimée par la formule suivante (en négligeant le temps du verbe) :

AIMER (DEF (JEUNE (ENFANT)), DEF (SOUPE))

Et une phrase plus complexe comme "Est-ce que les chevaux ne mangent pas de foin ?" sera exprimée par la formule :

. QU (NEG(MANGER (PLU (DEF (CHEVAL)), PART (FOIN))))

On voit ainsi que cette formulation permet de représenter le sens d'une phrase de manière assez commode. Mais on a déjà dit qu'on pouvait très bien envisager d'autres formulations et il faut bien préciser les caractéristiques de celle-ci. Une première remarque est que les relations dépendent de l'ordre des arguments et du parenthésage. Or les parenthèses sont lourdes à manipuler et il faut savoir pour un verbe donné par exemple

quel argument représente le sujet ou l'objet. On pourrait alors utiliser une autre représentation qui donnerait explicitement le cas de chaque argument. C'est une proposition de Fillmore qui a été suivie dans un certain nombre de systèmes en intelligence artificielle, et qui donne des résultats satisfaisants.

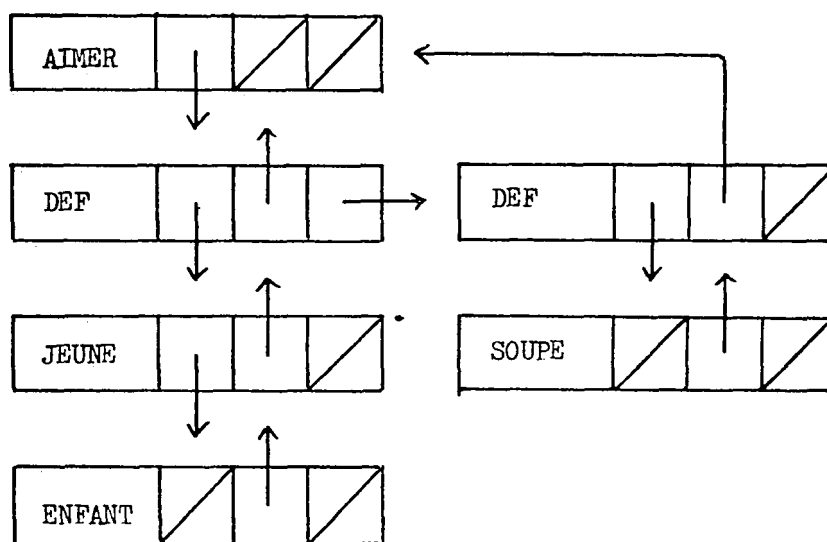
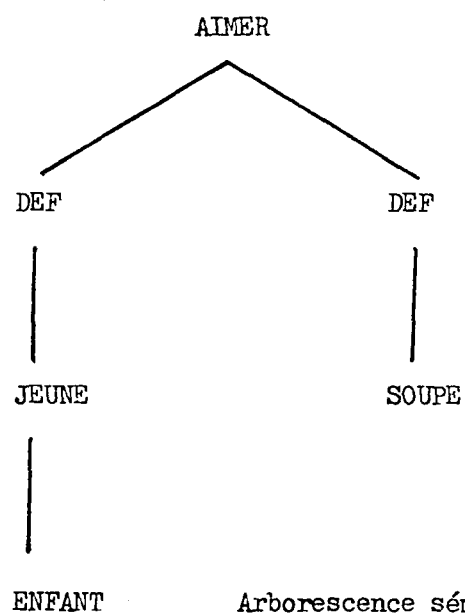
Mais on peut considérer ces différentes notations comme équivalentes dans la mesure où elles permettent en fait d'exprimer les mêmes choses. D'ailleurs les langues naturelles utilisent pour signifier les relations grammaticales tantôt l'ordre des mots (anglais, chinois) tantôt des désinences casuelles (latin, russe) et on trouve souvent un mélange de ces deux moyens dans une même langue (allemand, japonais). Ce problème du choix entre notations équivalentes revient souvent en linguistique formelle, car on n'a pas de critère empirique de décision qui soit vraiment convaincant.

Ensuite on a vu que les éléments sémantiques utilisés ici étaient le plus souvent très proches des mots ordinaires de la langue naturelle. On n'essaye donc pas de décomposer le sens d'un mot en primitives sémantiques plus simples que les mots du lexique. Car les tentatives faites dans ce sens, par Schank par exemple, sont intéressantes mais difficiles à justifier, et cela aurait beaucoup compliqué le processus de génération.

On n'a pas non plus cherché à associer une estimation continue ou multi-valuée aux relations représentées. Ce genre de valuation permettrait un calcul de plausibilité et des raisonnements flous, mais on n'a pas traité le problème. De même on s'est limité à ne générer qu'une phrase unique par formule sémantique. On a laissé de côté la question des variantes stylistiques et des paraphrases, qui ne sont pas nécessaires en génération (mais qu'il faut savoir reconnaître à l'analyse).

De plus, il faut bien voir que l'aspect formel des énoncés ne devrait pas faire croire qu'on peut les assimiler sans précautions aux formules de la logique classique. La parenté est réelle et délibérée pour pouvoir s'inspirer des résultats importants obtenus par les logiciens. Mais les énoncés sémantiques sont loin d'être défini avec la même rigueur qu'un langage logique, et on n'a pas essayé de leur donner une évaluation. Et on a négligé des problèmes importants comme l'utilisation des quantificateurs, la coordination, l'usage de modalités...

AIMER (DEF (JEUNE (ENFANT)), DEF (SOUPE))



Représentation en machine

.../...

3-3 Arborescence sémantique

Ceci dit, après avoir choisi cette formulation, il faut enfin lui donner une représentation en machine. Or il est commode de travailler sur un graphe plutôt que sur un énoncé linéaire, et le graphe correspondant à ce type de formule est une arborescence. De tels graphes se représentent très bien par des structures de listes, mais il y a plusieurs conventions possibles. Ici encore on remarquera que ces différentes représentations sont tout à fait équivalentes et le passage d'une représentation à une autre ne change rien au contenu.

La représentation la mieux adaptée au traitement à effectuer sera donc la meilleure, et on a choisi une convention particulière pour des raisons pratiques. Ainsi on construira l'arborescence sémantique en faisant pointer chaque noeud sur le "fils gauche" et le "frère droit" du noeud (voir figure), car cela permet d'avoir un nombre quelconque de successeurs à tout noeud. On fait aussi pointer chaque noeud sur son "père" pour pouvoir remonter l'arborescence. De la sorte, on peut représenter un nombre quelconque d'arguments par prédicat, et on dispose d'algorithmes simples pour le parcours de l'arborescence.

Ainsi cette arborescence est l'objet représentant le sens du message en machine. Il reste à engendrer la phrase correspondante en langage naturel, et pour cela il faut un dictionnaire et une grammaire.

4) DICTIONNAIRE

On vient de voir que les éléments sémantiques choisis sont généralement proches des mots naturels du français. Il suffira donc lors de la génération de faire correspondre un mot français à chaque élément de sens (sauf s'il doit se traduire par un processus grammatical comme l'interrogation ou le pluriel). Mais les caractéristiques grammaticales et la prononciation du mot exprimant une notion donnée sont assez imprévisibles et varient d'ailleurs largement d'une langue à l'autre. Il faut alors disposer d'un dictionnaire contenant ces informations pour pouvoir construire des phrases françaises grammaticalement acceptables.

.../...

4-1 Entrées lexicales

Ce dictionnaire se compose d'entrées lexicales qu'il aura fallu fournir préalablement au programme de génération. Chaque entrée indique , pour une notion sémantique, la catégorie grammaticale, les caractéristiques morphologiques et syntaxiques, et la prononciation du mot français correspondant . Dans le cas des mots variables (verbes, adjectifs...) on donne en prononciation une racine dont on puisse déduire toutes les formes fléchies. De plus chaque entrée comporte un pointeur pour renvoyer à des formes supplémentaires éventuelles, qui sont nécessaires pour la conjugaison des verbes irréguliers .Ceci permet de gagner de la place sur les mots réguliers, puisqu'on ne réserve pas d'espace ^{genre} pour les formes irrégulières. Un autre pointeur assure le chainage des entrées lexicales dans le dictionnaire. 7

Ainsi une entrée indiquera par exemple si un mot est un nom ou un adjectif ou un verbe. Si c'est un nom, le dictionnaire donnera aussi son genre et dira s'il a un pluriel irrégulier. Pour un adjectif, il donnera la formation du genre et la position antérieure ou postérieure au nom qualifié ; pour un verbe, la conjugaison et la construction syntaxique. En fait, on s'est limité aux caractéristiques essentielles, car la construction des verbes en particulier pose des problèmes complexes. A ce sujet, nous renvoyons aux travaux de M.Gross et son équipe. Quant aux éléments sémantiques "grammaticaux" comme la négation ou l'article, ils ne se trouvent pas dans le dictionnaire où ils seraient inutiles.

4-2 Structure du dictionnaire

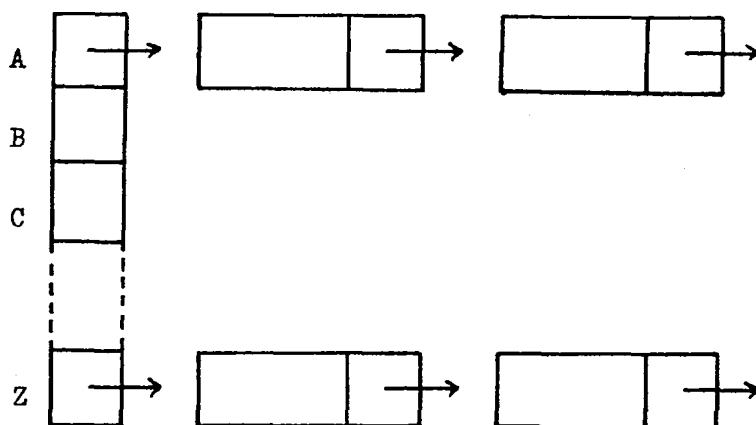
Pour le dictionnaire aussi on utilise une structure de liste pour pouvoir le modifier facilement à tout moment. Les entrées lexicales sont chaînées par pointeur et on peut donc ajouter, remplacer ou supprimer une entrée quelconque sans avoir à se soucier de l'emplacement en mémoire. Ainsi quand on veut ajouter une entrée, il suffit de la fournir en donnée au programme, qui l'insère automatiquement dans le dictionnaire à la bonne place dans l'ordre alphabétique.

C'est d'ailleurs de cette manière qu'on construit le dictionnaire préalablement à la génération. On donne une première fois au programme les entrées lexicales dans un ordre quelconque, et elles sont insérées en liste dans l'ordre alphabétique, même si le dictionnaire

SEMANTIQUE	CATEGORIE	MORPHOLOGIE	SYNTAXE	PHONETIQUE	POINTEURS
------------	-----------	-------------	---------	------------	-----------

SOUPE	Nom	Régulier	Féminin	SOUP
JEUNE	Adjectif	Invariable	Antérieur	JEUN
AIMER	Verbe	1er Groupe	Transitif	AIM

Entrées lexicales



Structure du dictionnaire

est vide au départ. De fait, il n'y a pas de différence entre la création du dictionnaire et les modifications ultérieures, car on utilise exactement les mêmes procédures.

Cependant, le dictionnaire ne se constitue pas d'une liste unique chainant toutes les entrées. En fait, pour diminuer les temps d'accès lors de la construction ou de la lecture du dictionnaire il est divisé en 26 sous-listes, c'est-à-dire une section par lettre de l'alphabet. On recherche donc l'emplacement d'une entrée comme dans un dictionnaire ordinaire, en commençant par se reporter à la section correspondante à la première lettre de l'entrée considérée. Puis seulement on cherche dans l'ordre alphabétique dans la section idoine. On évite ainsi d'avoir à parcourir séquentiellement des listes trop longues.

Le dictionnaire a donc été conçu comme une structure extensible. On n'a pas cherché à construire un lexique important ni même à se limiter à une application particulière. Mais on a veillé soigneusement à ce que la forme des entrées lexicales, la structure du dictionnaire et les règles de la grammaire permettent d'ajouter facilement n'importe quel mot au gré des besoins.

IV - GRAMMAIRE ORALE

Une fois construite l'arborescence sémantique, le programme doit maintenant à partir de cette arborescence générer la phrase française correspondante. Cela se fera en consultant le dictionnaire rentré au préalable, mais il faut aussi disposer de règles de grammaire. En effet il faut mettre les mots variables à la forme demandée par leur fonction (morphologie), placer les mots dans le bon ordre (syntaxe) , puis assurer les modifications dues à l'enchaînement (liaisons).

On a donc mis au point des règles purement orales tout comme le dictionnaire, l'ensemble constituant une grammaire de base du français parlé, assez différente de la grammaire de la langue écrite. C'est d'ailleurs la réalisation de ces règles morpho-syntaxiques orales qui a demandé le travail le plus important.

1) MORPHOLOGIE

Dans beaucoup de langues la forme des mots est variable selon leur fonction dans la phrase, et l'étude de ces variations constitue la morphologie. A cet égard, le français comme toutes les langues romanes possède une morphologie riche, surtout pour les verbes et adjectifs. Certes la morphologie française est plus simple que celle de langues à déclinaisons comme le latin ou le russe, mais plus compliquée qu'en anglais ou en chinois. Il faudra donc déterminer les facteurs qui commandent ces variations de forme (c'est-à-dire en français essentiellement les accords en genre, nombre et personne), puis donner en conséquence la bonne forme de chaque mot variable.

Or c'est justement en morphologie que le décalage entre français oral et écrit est le plus évident. Car les marques morphologiques se trouvant surtout en fin de mot ont subi une importante usure phonétique dans la langue parlée, alors que l'orthographe conservait les anciennes formes. Ainsi des formes verbales qui diffèrent dans la langue

.../...

écrite ont une prononciation identique ("aime, aime(s), aime(nt)", "fini(s), fini(t)"), le féminin oral de beaucoup d'adjectifs se forme en ajoutant une consonne et non pas un "e" ("peti (t), petit (e)"), le pluriel oral des noms est marqué par le déterminant et non par la terminaison ("l'homme, les homme (s)")... De même les liaisons si fréquentes en français parlé ne sont pas explicites dans l'orthographe.

Mais si cette morphologie orale est dans l'ensemble plus simple que la morphologie écrite (du fait de la perte de nombreuses finales), elle reste complexe. Et on peut dire qu'elle diffère autant de la forme écrite du français, que d'une autre langue romane voisine comme l'italien ou l'espagnol ! Or elle n'est pas décrite dans les grammaires traditionnelles.

Il a donc fallu réaliser d'abord un travail non négligeable de description linguistique avant de pouvoir programmer, car les études sur le français oral que nous avons pu consulter sont intéressantes mais fragmentaires (voir par exemple A. Rigault). Puis on a mis les connaissances ainsi acquises sous forme de procédures d'accord donnant la forme d'un mot en contexte. Mais on verra avec la syntaxe comment le contexte détermine l'accord, car les procédures morphologiques supposent que le genre, le nombre et la personne soient déjà connus.

1-1 Verbes

Les mots les plus variables en français sont les verbes, dont la conjugaison montre un grand nombre de formes (plusieurs dizaines !). En effet les verbes varient selon la personne, le nombre, le temps, le mode. De plus il y a un nombre important de verbes irréguliers. Et quand on essaye de décrire la conjugaison du point de vue oral, on est amené à modifier la classification traditionnelle. Certes les formes orales sont plus simples et moins nombreuses que les formes écrites et certains temps ne s'emploient plus dans la langue parlée (comme le passé simple, ou l'imparfait du subjonctif). Mais le système oral est assez différent pour demander une formulation neuve.

En fait on s'est limité au présent de l'indicatif. C'est la raison pour laquelle on n'indique ni le temps ni le mode dans les formules sémantiques ou dans la grammaire. Mais ce n'est pas une restriction fondamentale, parce que la conjugaison des autres temps et modes se fait selon les mêmes procédés (par suffixation), sauf les temps composés

qui emploient un auxiliaire. Il ne serait donc pas trop difficile d'étendre ce qu'on va exposer ici, du moins pour une partie de la conjugaison.

En tenant compte de tous les verbes français au présent de l'indicatif, on a pu les classer en cinq groupes différents selon la conjugaison : 3 groupes réguliers qui peuvent se conjuguer à partir d'un radical unique par verbe, et 2 groupes irréguliers où chaque verbe a plusieurs radicaux. Les groupes réguliers comprennent beaucoup plus de verbes que les groupes irréguliers, mais les verbes irréguliers sont très fréquents (c'est sans doute pourquoi ils n'ont pas été régularisés). Ces groupes sont les suivants :

- 1er groupe régulier : il correspond à peu près au 1er groupe écrit des verbes à infinitif en "er" ("marcher"), mais sans les verbes comme "peser", "jeter", "envoyer"... dont le radical change ("pèse mais "pesons", "envoie" mais envoyons") et qui doivent donc être classés avec les verbes irréguliers.
- 2ème groupe régulier : il correspond au 2ème groupe écrit des verbes à infinitif en "ir" , qui se prononcent en ajoutant un "i" au radical plus la consonne "s" aux personnes du pluriel ("finit", "finissent").
- 3ème groupe régulier : il comprend des verbes classés à l'écrit dans le 3ème groupe des verbes irréguliers, mais qui sont en fait parfaitement réguliers à l'oral. Ceux-ci perdent la dernière consonne du radical aux personnes du singulier au présent ("rend", "rendent").
- Groupe irrégulier : il contient les autres verbes du 3ème groupe écrit ("prendre" par exemple) et les verbes irréguliers du 1er groupe écrit ("peser"), dont le radical varie de façon difficilement prévisible. On pourrait y distinguer des sous-groupes, mais pour simplifier on donnera systématiquement dans le dictionnaire les trois formes du radical à la 1ère personne du singulier et aux 1ère personne et 3ème personne du pluriel ("prends", "pren-ons", "prennent").
- Groupe très irrégulier : il consiste en cinq verbes seulement ("être", "dire", "faire", "avoir", "aller"), mais qui sont tellement irréguliers qu'on donnera les six formes du présent en entier dans le dictionnaire ("suis", "es", "est", "sommes"...), plutôt que d'essayer de les conjuguer.

	<u>1er GROUPE</u>	<u>2ème GROUPE</u>	<u>3ème GROUPE</u>
Je	R	R+i	R
Tu	R	R+i	R
Il	R	R+i	R
Nous	R+ $\tilde{o}$	R+i+s+ $\tilde{o}$	R.C+ $\tilde{o}$
Vous	R+e	R+i+s+e	R.C+e
Ils	R	R+i+s	R.C

<u>Ex:</u>	"marcher"	"finir"	"rendre"
	R = mar $\int$	R = fin	R.C = r $\tilde{a}$.d

	<u>GROUPE</u> <u>IRREGULIER</u>	<u>GROUPE TRES</u> <u>IRREGULIER</u>
Je	A	A
Tu	A	B
Il	A	C
Nous	B+ $\tilde{o}$	D
Vous	B+e	E
Ils	C	F

<u>Ex:</u>	"prendre"	"être"
	A = pr $\tilde{a}$	A = syi B = e
	B = præn	C = e D = s $\tilde{o}$ m
	C = præn	E = ϵ t F = s $\tilde{o}$

Conjugaison orale

Il faut aussi mentionner les verbes pronominaux, qui, tout en appartenant à l'un des groupes ci-dessus, sont précédés d'un pronom réfléchi variant avec la personne ("me", "te", "se"...). Mais ce pronom reste séparable du verbe.

Bien sûr cette classification est discutable comme toute classification. Et il y a plusieurs façons possibles pour couper entre le radical et les terminaisons. Par exemple on pourrait considérer le "i" du 2ème groupe comme faisant partie du radical, même s'il est prévisible. On pourrait aussi trouver des règles d'alternances pour des sous-groupes de verbes irréguliers (comparer "peux", "pouvons" avec "veux", "voulons"), mais cela compliquerait la conjugaison. Le choix retenu est un compromis entre la complexité des règles et l'encombrement du dictionnaire.

Donc pour conjuguer un verbe donné, la procédure d'accord des verbes consulte le dictionnaire pour connaître les caractéristiques morphologiques et les radicaux du verbe. Puis grâce à ces renseignements lexicaux et en fonction de la personne et du nombre, la procédure met le verbe à la forme demandée par le contexte de la phrase. Mais il restera bien sûr à placer cette forme au bon endroit.

1-2 Adjectifs

La morphologie de l'adjectif est beaucoup plus simple, car il n'y a accord qu'en genre et en nombre. Mais contrairement au verbe, la morphologie orale est cependant plus complexe pour l'adjectif que ne le laisserait prévoir la forme écrite. Ainsi malgré de nombreuses exceptions et sous-règles, la règle orthographique est très claire : on ajoute un "e" au féminin et un "s" au pluriel (ainsi "grand", "grande", "grands", "grandes"). En fait ces désinences graphiques ne se prononcent pas elles-mêmes et les marques orales de genre et de nombre forment un système bien différent, dont l'orthographe masque le fonctionnement.

Par exemple beaucoup d'adjectifs qui varient en genre dans leur forme écrite sont invariables à l'oral ("joli", "jolie" ; "noir", "noire"). Beaucoup d'autres forment le féminin en ajoutant une consonne ("petit", "petite") à l'oral, avec souvent une modification de la voyelle précédente ("plein", "pleine" ; "sot", "sotte"). De plus des phénomènes de liaisons font varier encore les formes non seulement au pluriel ("grands garçons" mais "grands _ enfants") mais aussi au singulier

("grand garçon" mais "grand _enfant"). Et la liaison peut provoquer la dénasalisation de la voyelle précédente ("plein feu", mais "plein _air" prononcé comme "pleine"). Tout ceci n'apparaît pas dans l'orthographe.

Il apparaît ainsi que les variations orales suivent des règles assez compliquées, et si différentes de la forme écrite qu'il faut un traitement complètement phonétique. On a alors mis au point une classification orale et des règles orales. On s'est arrangé pour qu'il suffise d'une seule forme dans le dictionnaire pour prévoir toutes les autres. Mais cette forme de base est celle du féminin, et on dérivera le masculin du féminin contrairement à la formulation traditionnelle.

En effet pour les nombreux adjectifs dont le féminin est marqué par une consonne, cette consonne n'est pas prévisible ("petit", "petite" mais "grand", "grande") alors qu'il est facile de l'effacer pour obtenir le masculin. De même on peut prévoir les formes de liaison à partir du féminin seulement, ainsi que les alternances orales/nasales ("plein ", "pleine" mais "fin", "fine") auxquelles les voyelles nasalisées donnent lieu.

Le pluriel étant beaucoup plus régulier, la classification des adjectifs est basée essentiellement sur les particularités de la formation du genre oral. Cette classification est donc la suivante :

. Genre invariable : ces adjectifs ("joli", "calme") ne posent aucun problème, sauf quelques uns dont le masculin pluriel est marqué par le passage d'une finale /al/ à /o/ ("amical", "amicaux") et qui doivent être signalés dans le dictionnaire.

. Substitution de la finale : un certain nombre changent de finale suivant quelques règles simples. Ainsi à partir de la forme du féminin : /-v/ donne /-f/ ("vive", "vif"), /-øz/ donne /-ær/ ("moqueuse", "moqueur"), et /-tris/ donne /-tær/ ("motrice", "moteur"). Noter aussi "sèche", "sec".

. Adjonction d'une consonne : beaucoup d'adjectifs ajoutent une consonne au féminin. Il faut l'enlever au masculin mais la conserver en cas de liaison ("petite", "petit" mais "petit _enfant") tout en la modifiant parfois ("grande" mais "grand _homme"). De plus il faut souvent modifier la voyelle précédente ("pleine", "plein") sauf quelquefois en liaison. Les règles sont ici assez compliquées....

	<u>MASCULIN</u>		<u>FEMININ</u>	
	Singulier	Pluriel	Singulier	Pluriel
<u>INVARIABLE</u>				
"noir"	nwar	nwar(z)	nwar	nwar(z)
"amical"	amikal	amiko(z)	amikal	amikal(z)
<u>SUBSTITUTION</u>				
"vif"	vif	vif(z)	viv	viv(z)
<u>ADJONCTION</u>				
"petit"	pəti(t)	pəti(z)	pətit	pətit(z)
"grand"	grɑ̃(t)	grɑ̃(z)	grɑ̃d	grɑ̃d(z)
"sot"	so(t)	so(z)	sɔt	sɔt(z)
"plein"	plɛ̃ (plɛn)	plɛ̃(z)	plɛn	plɛn(z)
<u>IRREGULIER</u>				
"beau"	bo (bɛl)	bo(z)	bɛl	bɛl(z)

(les parenthèses indiquent les formes de liaison devant voyelle)

Morphologie orale de l'adjectif

. Genre irrégulier : un petit nombre perdent aussi la consonne finale au masculin, mais avec un changement radical de la voyelle précédente ("belle", "beau"). La forme du féminin est aussi la forme de liaison du masculin ("bel homme").

Mais on pourrait considérer cette dernière catégorie comme un cas particulier de la précédente. De même on peut scuter que certaines alternances vocaliques ("sot", "sotte" ou bien "moqueur" "moqueuse") sont dues à des règles générales de la phonologie du français standard et ne sont pas limitées aux adjectifs. Les règles adoptées sont donc discutables, mais elles couvrent tous les cas recensés.

Enfin au pluriel et quelle que soit la catégorie, on ajoutera en dernier la consonne /z/, mais seulement en cas de liaison (donc devant un nom commençant par une voyelle). On voit que la morphologie de l'adjectif dépend en grande partie des liaisons. C'est à dire que la forme d'un adjectif ne peut être complètement fixée avant qu'il ne se trouve en position définitive dans la phase phonétique. La touche finale du traitement sera donc assurée avec les liaisons en général, qu'on verra en détail un peu plus loin. Ceci dit, on peut remarquer que l'orthographe permet aussi souvent de prévoir les alternances.

1-3 Autres mots

Les autres mots variables en français sont les noms, les déterminants en général et les pronoms. Mais la morphologie du nom est très réduite, et pronoms et déterminants peuvent être mis à part car ce ne sont pas des mots lexicaux et leur morphologie n'est donc pas productive.

Le pluriel des noms est le plus souvent marqué par un "s" dans la forme écrite, mais au contraire la forme orale est généralement invariable. Car pour des raisons syntaxiques il n'y a pratiquement jamais de liaison avec un mot suivant. Seuls quelques noms irréguliers ont un pluriel marqué en /o/ ("cheval", "chevaux" ; "travail", "travaux") ou bien complètement imprévisible ("œil", "yeux"). Ces exceptions devront être signalées dans le dictionnaire. Quant au genre, il est inhérent au nom mais doit être indiqué dans le dictionnaire pour l'accord des déterminants et adjectifs.

Les déterminants (articles, démonstratifs, possessifs...) forment une petite classe fermée servant à actualiser le groupe nominal. Ils s'accordent en genre et en nombre, mais leur morphologie est limitée. Et en fait on n'a traité que les articles, définis ("le", "la", "les"), indéfinis ("un", "une", "des") et partitifs ("du", "de la", "des"). Mais il ne serait pas difficile de traiter d'autres déterminants de la même manière. Faisant partie du groupe nominal comme tous les déterminants, les articles sont sujet à la liaison ("les_enfants") mais aussi à l'élision ("le garçon" mais "l'homme"). De plus ils fusionnent avec des prépositions dans certains cas ("à la femme" mais "au garçon"). Ces liaisons, élisions et contractions seront détaillées plus loin, car ces phénomènes ne sont pas particuliers aux articles. Enfin les articles dépendent aussi de la négation ("je bois du vin" mais "je ne bois pas de vin").

Les pronoms forment aussi une classe fermée, mais ils ont un assez grand nombre de formes, car en plus des variations de personne, de genre et de nombre, ce sont les seuls mots du français à avoir conservé des formes différentes selon les cas (sujet, objet...) comme en latin. Ainsi les pronoms personnels peuvent présenter jusqu'à cinq cas différents : sujet ("ils"), objet ("les"), objet indirect ("leur"), disjoint ou après préposition ("eux"), réfléchi ("se"), même s'il y a aussi des formes non différenciées ("nous"). De plus ils peuvent subir la liaison ("vous_allez") et l'élision ("on la voit" mais "on l'appelle"). Et le placement des pronoms personnels, le plus souvent devant le verbe, obéit à des règles complexes (par exemple : "tu ne le lui donne pas" mais "tu ne me le donne pas").

On n'a donc traité les pronoms que partiellement, à l'intérieur du groupe verbal. Il faut noter qu'on a considéré "nous" et "vous" comme des personnes distinctes (plutôt que comme le pluriel de "je" et "tu"), car il n'est pas certain que "nous" représente plusieurs "je", et "vous" sert aussi souvent de personne de politesse en français. Selon cette analyse, le 3ème personne serait ainsi la seule à présenter des variations de genre et de nombre, mais c'est un choix qu'on pourrait discuter.

2) SYNTAXE

Comme on vient de le voir, la morphologie du français est assez compliquée, mais ce n'est pas une langue à cas. L'ordre des mots, c'est-à-dire la syntaxe, a donc une grande importance pour exprimer

.../...

les relations grammaticales (sujet, objet, déterminant...). Or la syntaxe du langage des énoncés sémantiques est différente de celle du français, et il faudra alors placer les mots dans l'ordre de la langue naturelle pour la génération de phrases.

Mais les relations sont aussi marquées par les accords morphologiques en liaison étroite avec la syntaxe, car l'accord ne se fait que dans des contextes syntaxiques bien précis. C'est pour cela qu'on a considéré les accords comme faisant partie de la syntaxe au sens large. On accorde donc les mots variables avant de les placer. Enfin la structure syntaxique se manifeste aussi dans la prosodie, et il faudra introduire dans les phrases des marqueurs pour la synthèse de la prosodie.

2-1 Accords

En français comme dans beaucoup de langues, il y a accord entre les mots variables en relation grammaticale étroite : verbe avec le sujet, déterminants et adjectifs avec le nom, etc... Dans la morphologie française, l'accord se fait en genre, nombre et personne pour déterminer la bonne forme d'un mot dans le contexte de la phrase. La morphologie est donc difficilement séparable de la syntaxe, et les relations grammaticales s'expriment de façon redondante à la fois par l'ordre et par l'accord.

Ce système mixte est une survivance d'un état de langue antérieur (le latin) où l'ordre des mots était beaucoup plus flexible, car les relations grammaticales étaient marquées essentiellement par des variations de forme. De même les pronoms personnels ont conservé une flexion casuelle alors que leur syntaxe rigide suffit à distinguer leurs fonctions.

Mais on ne peut pas accorder un mot variable avant d'avoir trouvé le nom ou le pronom auquel il se rapporte. Or dans les énoncés sémantiques comme dans la langue naturelle, les noms ou pronoms peuvent se situer après les mots qui s'accordent avec eux. Il faudrait donc revenir en arrière pour accorder un adjectif par exemple après avoir rencontré le nom suivant, et les retours en arrière ne sont pas faciles à programmer. On a préféré parcourir l'arborescence sémantique en deux passes, la première ne servant qu'à déterminer les genres, nombres et personnes nécessaires aux accords, en consultant le dictionnaire. On empile ces indications et on reste pointé sur le dictionnaire.

.../...

Plus précisément on empile un genre, un nombre et une personne, pour chaque colonne de l'arborescence (c'est-à-dire tant qu'on ne rencontre pas de parenthèse fermante dans l'énoncé sémantique). En effet, la représentation choisie est telle qu'il n'y a qu'un seul nom ou pronom par colonne et que tous les mots variables de la colonne s'accordent avec ce dernier. La portée des accords est ainsi automatiquement déterminée par la convention de représentation, ce qui n'est pas évident à priori.

Puis on parcourt à nouveau l'arborescence en une deuxième passe, et on dépile les indicateurs d'accord (déterminés lors de la première passe) au fur et à mesure qu'on passe d'une colonne à une autre. On peut alors effectuer les accords en genre, nombre et personne en faisant appel aux procédures morphologiques qu'on a vues plus haut, et grâce aux indications du dictionnaire sur les particularités de chaque mot. Il faudra ensuite placer les mots accordés dans l'ordre de la phrase.

Ainsi en anticipant un peu sur le placement syntaxique, on obtiendra par exemple les formes suivantes en phonétique :

"Le bon vin" à partir de DEF (BON (VIN))

mais "La bonne soupe" à partir de DEF (BON (SOUPE))

De même DEF (BEAU (OISEAU)) donne "Le bel oiseau"

mais PLU (DEF (BEAU (OISEAU))) donnera "Les beaux oiseaux"

Ou bien encore pour illustrer l'accord du verbe en personne :

BOIRE (LUI) devient "Il boit"

mais BOIRE (VOUS) donne "Vous buvez"

et bien sûr, toujours sous forme orale (et non pas écrite)

dans le code choisi pour la synthèse de parole.

2-2 Placement

C'est aussi lors de la deuxième passe sur l'arborescence qu'on placera les formes phonétiques (tirées du dictionnaire, du moins pour les mots lexicaux, et accordées si nécessaire) dans l'ordre demandé par la syntaxe française. Pour cela on crée une nouvelle liste linéaire où on place les mots de façon à pouvoir en modifier l'ordre à tout moment au fur et à mesure qu'on construit la phrase à partir de l'arborescence.

Une fois de plus on aura ainsi utilisé les structures de listes pour la souplesse des modifications qu'elles permettent.

Par exemple on insère le groupe sujet avant le verbe, alors qu'il vient après le verbe dans l'arborescence sémantique, mais on ajoute le groupe objet à la suite :

ainsi AIMER (DEF(JEUNE(ENFANT)), DEF (SOUPE))

donne "Le jeune enfant aime la soupe" (en code phonétique)

La syntaxe du groupe verbal est particulièrement compliquée, car les différents types de pronoms personnels se placent en général devant le verbe, mais la négation est insérée en partie après le pronom sujet et en partie après le verbe :

NEG (S'ARRETER (TOI)) donne : "Tu ne t'arrêtes pas"

Mais on est loin d'avoir traité complètement dans ce programme le placement des pronoms personnels, qui pose des problèmes difficiles. Par exemple on dit selon le verbe :

"Je pense à Jean" et "Je pense à lui"

mais "Je parle à Jean" et "Je lui parle"

Et en fait au moment d'écrire ceci le programme ne génère correctement que les pronoms sujets et les formes réfléchies des verbes pronominaux. Il reste donc évidemment à étendre le traitement aux autres formes des pronoms.

Par contre le verbe peut être transitif ("Il mange une pomme"), pseudo-intransitif ("Il mange"), intransitif ("Il dort"), et aussi transitif indirect exigeant une préposition "à" ou "de" selon le verbe devant l'objet ("Je pense à Jean"). Tous ces cas sont traités, à condition qu'il n'y ait qu'un seul objet. Les différentes constructions d'un verbe ("parler à", "parler de") sont considérées comme des entrées lexicales différentes, sauf pour l'emploi pseudo-intransitif d'un verbe transitif. Ainsi la préposition est introduite devant l'objet indirect :

PENSER (NOUS, DEF (SOUPE)) donne : "Nous pensons à la soupe"
mais il est possible de ne pas avoir d'objet du tout :
et PENSER (NOUS) donne simplement : "Nous pensons"

On obtient des phrases interrogatives en plaçant "est-ce que" en tête de phrase et en utilisant une intonation montante, du moins pour l'interrogation totale à réponse par oui ou non :

QU (NEG (BOIRE(PLU(ELLE), PART (VIN))))

donne : "Est-ce qu'elles ne boivent pas de vin ?" Cette tournure à l'avantage de ne pas changer l'ordre des mots.

Bien entendu les phrases sont générées en code phonétique, avec des marqueurs prosodiques pour calculer l'intonation. C'est seulement pour faciliter la lecture que ces quelques exemples ont été donnés sous la forme orthographique ordinaire.

Le programme peut donc générer des phrases simples, affirmatives, négatives, interrogatives, et interro-négatives, du type sujet-verbe-objet. Sujet et objet peuvent comporter article, adjectif et nom, ou pronom (dans certains cas). Le verbe peut avoir un objet direct ou indirect, ou bien être intransitif ou employé intransitivement. Ce n'est bien sûr qu'une petite partie de ce que serait une grammaire du français parlé, mais cela constitue un noyau minimum pour obtenir des messages élémentaires mais parfaitement normaux. Et comme la morphologie comprend aussi les formes irrégulières il n'y a pas de limitation lexicale, d'autant plus que les mots irréguliers sont en général les plus fréquents de la langue.

Cependant, il est bien évident que la syntaxe réellement utilisée, même dans le discours le plus fruste, est beaucoup plus compliquée, et demanderait un travail bien plus important. Mais cela aurait dépassé les limites de cette étude, et nous n'avons pas de prétentions sur ce sujet. Ensuite, contrairement à la morphologie où la forme orale n'est pas réalisée de la même façon que la forme écrite, la syntaxe parlée ne semble pas fondamentalement différente de la syntaxe écrite. Il s'agit plutôt d'une différence de style, de la fréquence plus ou moins grande des tours syntaxiques selon les cas, et on devrait pouvoir utiliser pour la langue parlée les travaux considérables accomplis sur la syntaxe écrite par les grammairiens et linguistes.

Il faut noter pourtant dans la langue parlée la rareté relative des propositions subordonnées, la reprise presque systématique du sujet ou de l'objet par des pronoms ("Et toi, tu le connais, ce type ?") la prépondérance de l'intonation pour marquer l'interrogation, etc... De plus le grand nombre de phrases inachevées et les changements de construction en cours de phrase demanderaient beaucoup de souplesse pour un système de reconnaissance, puisque les phrases sont souvent mal formées. Mais c'est moins important en génération, où on peut choisir la syntaxe qu'on désire utiliser et s'y limiter, comme on l'a fait ici.

Mais si on voulait obtenir une syntaxe plus complète, la structure du programme ne s'y prêterait pas facilement. Car les règles nécessaires sont nombreuses et dépendent les unes des autres. Aussi quand on veut rajouter une nouvelle règle, on est souvent obligé de modifier les règles déjà existantes. Ce n'est pas trop grave tant que la syntaxe reste limitée, mais les modifications deviennent vite inextricables si le nombre des règles augmente. Alors que la morphologie est traitée par des procédures faciles à modifier, la syntaxe du programme n'est donc pas aisément extensible. C'est un problème gênant, d'ailleurs très fréquent en programmation, surtout en intelligence artificielle où on doit utiliser des informations particulièrement complexes.

Il faudrait alors utiliser une autre structure mieux adaptée. Par exemple une structure d'automate permettrait de fournir les règles de syntaxe en données dans des tables, ce qui allègerait la programmation et faciliterait grandement la mise au point et l'évolution de la syntaxe. Pour poursuivre ce travail, nous remplacerions donc par un automate la partie du programme qui traite la syntaxe, avant de chercher à introduire de nouvelles règles.

2-3 Marqueurs prosodiques

La prosodie (intonation, durée, intensité) dépend du sens général de la phrase (affirmation ou interrogation par exemple), mais aussi de la structure syntaxique. En effet, l'allure globale de l'intonation peut marquer l'interrogation ou des nuances comme le doute, alors que les variations locales d'intonation et de rythme suivent les groupes syntaxiques. Sans entrer dans les détails, on notera que l'intonation monte à la fin des questions demandant une réponse par oui ou non, mais tombe normalement à la fin des autres types de phrases, et qu'elle monte puis redescend à la fin de chaque groupe syntaxique.

Si on faisait de la synthèse de parole à partir du texte écrit, il faudrait donc effectuer une analyse syntaxique plus ou moins complexe pour pouvoir calculer la prosodie, car la ponctuation ne donne que des indications insuffisantes. Mais ici le découpage syntaxique est connu lors de la génération et il suffit d'insérer dans la phrase phonétique des marqueurs appropriés pour délimiter les groupes prosodiques.

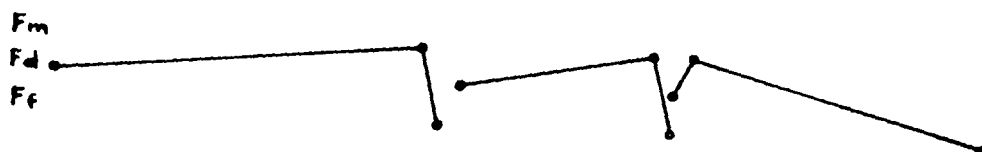
Texte

LES MEDICAMENTS SONT VENDUS EN PHARMACIE .

suite
phonétique

L) M) D I K A M * S / V * D U * F A R M A S i .

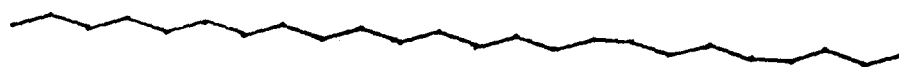
Intonation
de groupe



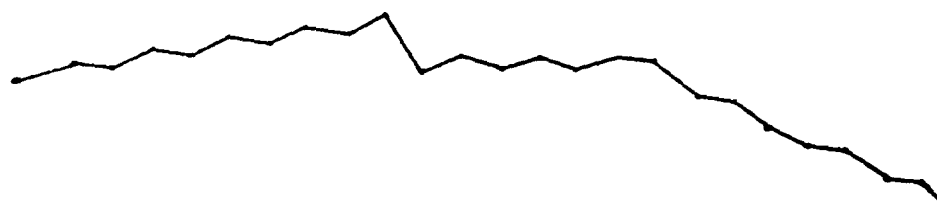
Intonation
de phrase



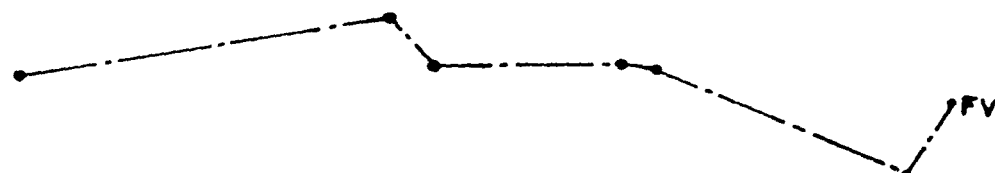
Micromélodie



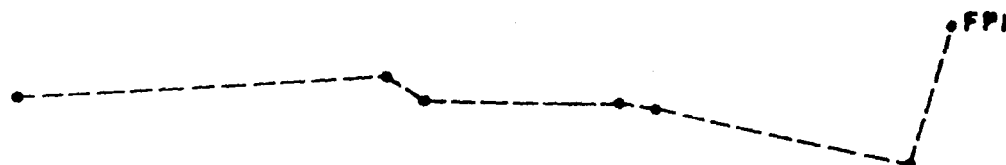
Intonation
finale
Cas .



Cas ,



Cas ?



Calcul de la mélodie

Ceci permettra de calculer facilement la prosodie, mais servira aussi à déterminer les liaisons, qui dépendent également de la structure syntaxique : on ne fait de liaison qu'à l'intérieur d'un groupe et jamais d'un groupe à l'autre, comme on verra plus loin.

Le verbe pouvant être considéré comme le pivot de la phrase, les marqueurs sont introduits lors du placement du groupe verbal. On insère un marqueur après le groupe sujet (sauf si c'est un pronom personnel, qui fait liaison avec le verbe), donc devant la négation et les pronoms éventuels, et un autre marqueur après le groupe verbal. En effet, la coupe avant le verbe paraît plus importante que celle qui suit le verbe, et il convient de les distinguer.

Enfin, si la phrase est une question oui-non, il faut ajouter un marqueur final pour indiquer l'intonation montante caractérisant l'interrogation.

Ainsi on aura par exemple les découpages prosodiques suivants :

"Le jeune enfant// aime / la soupe"

mais "Nous_aimons/la soupe" (le sujet est un pronom)

et "Les enfants// ne boivent pas/ de vin"

3) LIAISONS

La forme des mots varie selon leur fonction, mais dépend aussi de l'enchaînement des mots dans la phrase. C'est à dire que certaines variations morphologiques sont phonologiquement conditionnées par les mots voisins (les grammairiens emploient parfois le terme technique de "sandhi"). De tels phénomènes sont très fréquents en français parlé : ce sont les liaisons et élisions, et on peut y ajouter les formes contractées de l'article.

Nous les avons groupés sous le terme général de "liaison" pour simplifier l'exposé, car ces diverses variations ont en commun de dépendre de l'ordre final des mots. On ne peut donc les traiter qu'en dernier, après la construction de la phrase. Mais les liaisons dépendent aussi de la structure syntaxique, dont les marqueurs prosodiques gardent heureusement la trace. L'agencement même du programme permet alors d'adopter des solutions simples sans demander d'analyse supplémentaire.

3-1 Liaisons consonantiques

Ce sont les liaisons proprement dites : de nombreux mots ont deux formes différentes selon que le mot suivant commence ou non par une voyelle. La forme longue se termine par une consonne supplémentaire et s'emploie devant voyelle (dans le même groupe syntaxique), alors que la forme courte s'emploie partout ailleurs. On dit par exemple "petit_enfant" avec liaison, mais "peti (t) garçon" ou bien "il est peti (t)" sans consonne finale.

On voit que l'orthographe n'indique pas quand on fait la liaison ! D'autre part, la consonne de liaison varie d'un mot à l'autre et n'est pas prévisible à partir de la forme orale courte. Il faut donc décider dans quelles conditions on fait la liaison, et aussi déterminer la forme de liaison d'un mot donné.

En fait l'usage est assez variable, car les liaisons dépendent dans une certaine mesure du niveau de style. Plus la conversation est familière, moins on fait de liaisons, et beaucoup sont donc facultatives ("asse(z) aimable" ou bien "assez_aimable"). Pourtant certaines liaisons se font presque toujours et on a décidé de les considérer comme obligatoires.

Ces liaisons ont lieu régulièrement dans des groupes syntaxiques précis : groupe nominal déterminant-adjectif-nom ("le petit_enfant", "les_amis"), groupe verbal pronom-verbe ("nous_écrivons", "ils_aiment"), pour ne citer que les cas traités par le programme. Mais on pourrait aussi mentionner, entre autres, les liaisons après propositions ("sous_un arbre") ou après certains adverbes ("très_aimable").

Or justement le découpage syntaxique donné par les marqueurs prosodiques délimite bien les groupes de liaison. Il suffit donc de faire la liaison à l'intérieur de ces groupes, et de l'interdire devant les marqueurs, qui indiquent le passage d'un groupe au suivant. Ce découpage somme toute assez simple permet ainsi de déterminer correctement les cas de liaison aussi bien que la prosodie :

"Les petits_enfants//aiment/ la soupe"
 mais "Nous_aimons / le vin"

Mais il faut aussi interdire la liaison devant certains mots dans tous les cas, bien qu'ils commencent par une voyelle phonétique. En effet, alors qu'on fait normalement la liaison dans "les_amis", "les_hommes", on ne la fait pas dans "les hanches", "les hublots"... On dit traditionnellement de ces mots qu'ils commencent par un "h" aspiré, mais en fait ce "h" est parfaitement muet et rien (sinon l'étymologie) ne permet de prévoir leur comportement, d'autant plus que l'orthographe est incohérente ("les_hommes" avec liaison).

Il faut donc marquer ces mots dans le lexique, et on a tout simplement écrit leur forme phonétique dans le dictionnaire avec un "H" initial qui bloquera les liaisons, puis sera effacé (car il n'y a pas de phonème/h/correspondant en français).

Ainsi au cours de la génération 'L)Z.HUBLO', qu'on peut considérer comme une forme abstraite, donnera 'L) HUBLO' puis 'L) UBLO' ("les hublots") sans liaison. Mais le mot "homme" sera transcrit '+M' dans le code utilisé et donnera immédiatement 'L)Z +M' ("les_hommes"), avec liaison. De même on marquera par un "H" initial les mots commençant par une semi-voyelle qui ne font pas la liaison, comme par exemple "les yaourts" alors qu'on dit "les_yeux" avec liaison. Mais ce sont des cas plutôt rares.

Il reste à établir la forme de liaison des mots, c'est-à-dire essentiellement la consonne finale, mais avec parfois une modification de la voyelle précédente (ainsi "beau garçon" mais "bel homme"). Comme cette consonne de liaison est très variable selon les mots, il est évident que la forme de base doit être la forme longue avec consonne. Mais on signale cette consonne par un marqueur spécial, de façon à pouvoir l'effacer ailleurs que devant voyelle.

Ainsi pour les pronoms et les articles par exemple, on partira de la forme longue, et 'NWZ.' ("nous") donnera 'NWZ' devant voyelle et 'NW' sans liaison ailleurs. Noter que la forme de base se rapproche de l'orthographe, qui représente la prononciation ancienne.

Mais pour les adjectifs le problème est plus compliqué. Car on peut prédire la forme de liaison à partir du féminin de l'adjectif (mais pas à partir du masculin). Et dans de nombreux cas la forme de liaison est simplement identique phonétiquement à la forme du féminin : "petit_homme" comme "petite fille", "plein_air" prononcé comme "pleine".

.../...

Comme on avait justement choisi le féminin comme forme de base dans le dictionnaire, il suffit alors de prendre cette forme pour la liaison. Mais il faut aussi parfois modifier la consonne finale du féminin : "S" donnera "Z" ("grosse" mais "gros_homme") et "D" donnera "T" ("grande" mais "grand_homme").

De plus s'il n'y a pas de liaison, on doit dans certains cas modifier aussi la voyelle précédente avant d'effacer la consonne finale pour obtenir la forme isolée du masculin. Ainsi il faut nasaliser les voyelles devant "N" ("pleine" mais "plein"). Ce sont également ces formes, qu'on aura au masculin pluriel ("pleins", "beaux") et enfin dans tous les cas il faut prévoir une liaison possible en "Z" au pluriel des adjectifs : "beaux_enfants" comme "petits_enfants", "grandes_amies"...

Mais nous renvoyons pour une vue d'ensemble aux pages sur la morphologie de l'adjectif, car on vient de voir que le traitement des liaisons n'est pas simple, et aboutit à un nombre non négligeable de formes différentes.

3-2 Elisions et contractions

L'élision est un phénomène symétrique de la liaison : certains mots ont deux formes selon ce qui suit, comme pour les liaisons, mais la forme longue se termine par une voyelle qui disparaît devant voyelle. Ainsi on dit "la bière" mais "l'eau" comme l'indique d'ailleurs l'orthographe. Ces mots sont un petit nombre de mots grammaticaux étroitement liés au mot suivant dans le même groupe syntaxique : articles ("le", "la"), pronoms personnels ("je", "me", "te", "le", "la", "se"), particules diverses ("ne", "de", "que", "si"...).

Le traitement de l'élision est donc parallèle à celui de la liaison. Ainsi on ne fera d'élision qu'à l'intérieur des groupes délimités par les marqueurs prosodiques, et on l'interdit devant marqueur ou bien "h" aspiré : "l'eau" mais "la houle", "l'homme" mais "le hublot". De même on prend la forme longue comme forme de base, et on la signale par un marqueur spécial pour pouvoir effacer la voyelle finale devant voyelle. Par exemple 'LA,BY(R) donnera bien 'LA BY(R) ("la bière") mais 'LA, O' donnera 'L O' ("l'eau") avec élision. En bref, on traite les liaisons et les élisions de façon tout à fait similaire, sauf que l'élision est un problème plus restreint.

.../...

Enfin, il faut s'occuper aussi des formes contractées de l'article après les prépositions "à" et "de". En effet alors qu'on dit "à la fille", on ne dit pas "à le garçon" comme on aurait pu s'y attendre, mais "au garçon" : l'article fusionne avec la préposition précédente. Ces formes fusionnées sont les suivantes :

"à" + "le" donne "au"	"à" + "les" donne "aux"
"de" + "le" donne "du"	"de" + "les" donne "des"

En fait les formes "au" et "du" ne se trouvent que devant consonne (ou "h" aspiré). Devant voyelle, "le" se maintient mais sous forme élidée : "du garçon" mais "de l'homme" et "à l'homme". Et les formes "aux" et "des" font la liaison devant voyelle : "aux hommes", "des hommes". Ici encore l'orthographe ne doit pas faire oublier la nature phonétique du phénomène.

Mais on doit remarquer que ces contractions n'affectent que des mots grammaticaux bien précis, prépositions et articles, et non pas simplement une forme phonétique donnée. Sinon "pas levé" pourrait donner "pauvé" comme "à le" donne "au" ! On a donc marqué les mots "à" et "de" dès leur placement dans la phrase (y compris dans l'article partitif "du", "de l'"). Puis on fait la contraction avec l'article suivant si c'est "le" (mais devant consonne) ou "les", après avoir fixé l'ordre des mots. Enfin on effectue les liaisons et élisions éventuelles, selon la même méthode que dans le cas général.

Il ne reste plus alors qu'à transmettre la chaîne phonétique au programme de synthèse qui commandera le synthétiseur pour prononcer la phrase générée.

Par ailleurs on doit aussi tenir compte de certaines restrictions sur les articles possibles après la préposition "de" qui ne peut être suivie par un autre "de" ni par "des". En effet on parle "d'un cheval", mais "de chevaux" au pluriel et non pas "de des chevaux" comme on pourrait s'y attendre. De même on boit "de la bière" mais on parle "de bière" et non "de de la bière". L'article dépend donc de la préposition précédente.

On aurait pu traiter ce phénomène d'une façon similaire aux contractions de l'article, en effaçant dans la chaîne les formes de l'article impossibles après la préposition "de". Mais pour simplifier le traitement on a préféré régler le problème dès le placement syntaxique, en ne générant que les formes permises de l'article selon la construction du verbe. De la même manière on obtient les bonnes formes de l'article après la négation : "je ne bois pas de vin". Mais on peut discuter pour savoir si le résultat est une préposition ou un article...

CONCLUSION

Après avoir essayé de donner une vue d'ensemble du traitement automatique du langage naturel en insistant sur le langage parlé, on a exposé en détail la réalisation d'un programme de génération de phrases phonétiques. Dans le cadre des recherches sur le dialogue homme-machine par la parole, le but de ce programme était de passer du sens, donné sous forme d'énoncés de type logique, à la prononciation en notation phonétique. Il convient maintenant d'en évaluer les résultats, mais aussi les limitations, et enfin de voir les perspectives d'améliorations qu'on pourrait y apporter.

1) RESULTATS

Dans l'ensemble on a bien atteint le but qu'on s'était fixé : le programme génère correctement des phrases phonétiques bien formées correspondant aux formules sémantiques entrées. Ces formules peuvent être données sur cartes perforées, ou bien tapées directement sur un clavier en conversationnel, et on a testé avec succès la prononciation immédiate des phrases phonétiques par un synthétiseur de parole, prosodie comprise. A condition de fournir le sens au programme, on passe donc bien comme prévu du sens au son.

Le programme peut générer des phrases françaises simples du type sujet-verbe-objet, affirmatives, négatives, ou interrogatives. Sujet et objet peuvent comporter article, adjectif et nom, ou pronom. La construction du verbe peut être transitive ou intransitive, ou avec un objet indirect. Le programme effectue les accords morphologiques, le placement syntaxique, le marquage prosodique, les liaisons, tout ce qui est nécessaire à la prononciation d'une phrase.

Et le dictionnaire et les règles morphologiques ont été conçus de telle sorte que toutes les formes irrégulières sont prévues. Il n'y a donc aucune limitation lexicale, ce qui permet de s'adapter à n'importe quel domaine nouveau en ajoutant simplement des mots au dictionnaire. Ainsi le programme peut générer n'importe quel message

.../...

simple même imprévu, grâce à la richesse combinatoire des règles linguistiques naturelles qu'on a utilisées.

Ensuite il était intéressant de mener à bien un traitement purement oral de la grammaire française, sans jamais passer par le texte écrit. A notre connaissance, c'est la première tentative systematique de description du français oral, ce qui a demandé un travail important. Et cela permet de résoudre de façon élégante des problèmes comme la prosodie ou les liaisons, qui sont malaisés à traiter lorsqu'on part du texte écrit, car ce sont des phénomènes essentiellement oraux mal indiqués par l'orthographe.

Du point de vue informatique, le programme comporte environ 600 instructions en PL/1. Quant au temps de calcul, il est de moins d'une seconde par phrase générée sur IBM 370/168. Mais l'encombrement mémoire dépend de la taille du dictionnaire utilisé, bien que l'exécution proprement dite demande moins de 100 K octets.

2) LIMITATIONS

Cependant, les résultats obtenus sont limités. D'abord il faut fournir au programme les énoncés sémantiques, puisqu'on a laissé de côté le problème difficile de la génération du sens au cours du dialogue. C'est évidemment une lacune importante, et la formulation sémantique choisie, en fait assez proche de la grammaire de surface, peut paraître discutable et immotivée. Nous ne pouvons pas prétendre avoir apporté grand chose de nouveau du point de vue sémantique, car on a au contraire essayé d'utiliser une représentation neutre pour laisser la possibilité d'une évolution ultérieure.

Ensuite, la grammaire orale du programme ne peut générer qu'un petit sous-ensemble du français, même si on a voulu inclure les structures fondamentales. La morphologie n'est pas complète, car on ne peut conjuguer les verbes qu'au présent, alors que la conjugaison française comporte un grand nombre de temps et de modes différents. On n'a pas achevé non plus la déclinaison des pronoms personnels. Mais il serait assez facile de terminer la morphologie, qui se constitue d'un ensemble fini de formes qu'on peut raisonnablement espérer générer complètement.

Par contre, la syntaxe d'une langue naturelle est beaucoup plus complexe que la syntaxe rudimentaire du programme, qui ne peut générer que des propositions indépendantes simples. De plus la structure du programme rendrait difficile l'introduction d'un nombre important de règles supplémentaires. C'est là un problème d'informatique sans portée théorique, mais qu'il faut signaler, car cela limite sérieusement la syntaxe utilisable.

Enfin on peut remarquer que bien que le dictionnaire soit aisément extensible, en pratique on a fait les essais avec un lexique réduit. On n'a pas cherché à constituer un dictionnaire important, mais simplement à en vérifier le bon fonctionnement pour tous les types de mots.

Par ailleurs, on laisse aux programmes de synthèse de la parole le soin d'assembler les formes acoustiques et de calculer la prosodie. Or dans l'état actuel des connaissances en phonétique, on ne sait pas encore très bien comment déterminer les profils prosodiques. Cela reste un problème fondamental, qui demanderait des analyses plus poussées.

3) PERSPECTIVES

Mais on pourrait remédier à ces divers défauts en poursuivant le travail déjà accompli. L'amélioration la plus simple serait d'achever la morphologie, et en particulier les autres temps de la conjugaison des verbes. Cela ne devrait pas poser de problème, car la morphologie dans le programme se trouve sous forme de procédures faciles à modifier et à étendre. Cependant les temps composés qui se conjuguent avec un verbe auxiliaire demanderaient de nouvelles règles de syntaxe parce que l'auxiliaire peut être séparé du verbe par la négation ou des adverbes divers.

Puis il faudrait mettre au point une syntaxe plus complexe avec des règles plus nombreuses. Or on a vu que la structure actuelle du programme s'y prêtait mal parce que les modifications nécessaires risquaient de devenir inextricables. On pourrait alors utiliser une structure différente, comme par exemple un automate, qui soit plus souple et extensible. Et on pourrait aussi avoir un dictionnaire plus étendu selon les applications.

Enfin il faudrait s'attaquer au problème de la génération des énoncés sémantiques pour compléter la chaîne de réponse d'un système de dialogue. En s'inspirant des systèmes déjà existants, cela demanderait de réaliser un programme capable de raisonner par déduction ou inférence logico-sémantique dans le contexte du dialogue. Ce serait un travail important, mais on pourrait alors espérer dialoguer par la parole avec l'ordinateur, au moins dans un domaine donné plus ou moins limité.

BIBLIOGRAPHIE

LINGUISTIQUE GENERALE

- E. BACH & R. HARM (eds.) Universals in Linguistic Theory (Holt, Rinehart & Winston, New York, 1968)
- L. BLOOMFIELD Language (Holt, Rinehart & Winston, New York, 1933)
- N. CHOMSKY Syntactic Structures (Mouton, La Haye, 1957), Aspects of the Theory of Syntax (M.I.T. Press, Cambridge, 1965) Language and Mind (Harcourt, Brace & World, New York, 1968)
- N. CHOMSKY & M. HALLE Sound Pattern of English (Harper & Row, New York 1968)
- F. DUBOIS-CHARLIER & M. GALMICHE La Sémantique Générative (Langages N°27, Didier-Larousse, Paris, 1972)
- C. FUCHS & P. LE GOFFIC Initiation aux Problèmes des Linguistiques Contemporaines (Hachette, Paris, 1975)
- H.A. GLEASON Introduction to Descriptive Linguistics (Holt, Rinehart & Winston, New York, 1961)
- M. GROSS Méthodes en Syntaxe (Hermann, Paris, 1975)
- Z.S. HARRIS Methods in Structural Linguistics (University of Chicago Press, 1951)
- R. JAKOBSON & M. HALLE Fundamentals of Language (Mouton, La Haye, 1956)
- J. LYONS Introduction to Theoretical Linguistics (Cambridge University Press, Londres, 1968) Chomsky (Fontana-Collins, Londres, 1970)
- A. MARTINET Eléments de Linguistique Générale (A. Colin, Paris, 1960)
- S.A. SCHANE Generative Phonology (Prentice Hall, Englewood Cliffs, N.J., 1973)
- D.D. STEINBERG & L.A. JAKOBOVITS (eds) Semantics (Cambridge University Press, Londres, 1971)
- N.S. TRUBETZKOY Grundzüge der Phonologie (T.C.L.P., Prague, 1939)

INTELLIGENCE ARTIFICIELLE

- P. BRAFFORT L'Intelligence Artificielle (P.U.F., Paris, 1968)
- E. CHARNIAK & Y. WILKS (eds) Computational Semantics (North-Holland, Amsterdam, 1976)
- D. COULON, D.KAYSER & al. Description Générale d'un Système de Réponse aux Questions (C.R.I.N., Nancy, 1977)
- E. DELAVENAY La Machine à Traduire (P.U.F, Paris, 1972)
- B.L. NASH-WEBER & R.C. SCHANK (eds) Theoretical Issues in Natural Language Processing (Cambridge, Mass., 1975)
- J. PITRAT Un Programme de Démonstration de Théorèmes (Dunod, Paris, 1970)
La Programmation Informatique du Langage (La Recherche, Paris oct. 1978)
- R.C. SCHANK Conceptual Information Processing (North-Holland, Amsterdam, 1975)
- G. SABAH Contribution à la Compréhension Effective d'un Récit (Thèse d'Etat, Université de PARIS VI, 1978)
- J. WEIZENBAUM Eliza : a Computer Program (A.C.M. vol.9 n°1, 1966)
- T. WINOGRAD Understanding Natural Language (Academic Press, New York, 1972)
- W.A. WOODS Augmented Transition Networks for Natural Language Analysis (Harvard University, Cambridge, 1969)

TRAITEMENT DE LA PAROLE

- W.A. AINSWORTH Mechanisms of Speech Recognition (Pergamon, Oxford, 1976)
- C. CHOPPY Introduction de la Prosodie dans la Synthèse Vocale Automatique (Thèse de Docteur Ingénieur, Université de PARIS VI 1977)
- P. DELATTRE Comparing the Phonetic Features of English, German, Spanish and French (J. Groos Verlag, Heidelberg, 1965)
- P.B. DENES & E.N. PINSON The Speech Chain (Bell Telephone Labs, 1963)
- G. FANT Acoustic Theory of Speech Production (Mouton, La Haye, 1960)

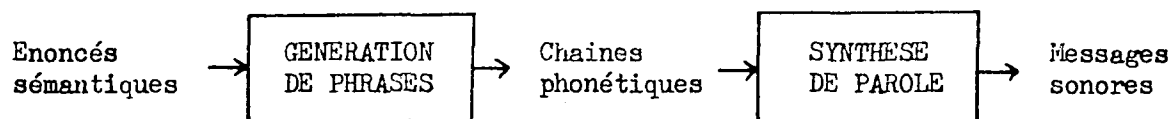
- J.L. FLANAGAN Speech Analysis, Synthesis & Perception (Springer-Verlag, Berlin, 1965)
- R. JAKOBSON, G. FANT & M. HALLE Preliminaries to Speech Analysis (M.I.T. Press, Cambridge, 1952)
- P. LADEFOGED Elements of Acoustic Phonetics (U. of Chicago Press, Chicago, 1962)
- I. LEHISTE (ed) Readings in Acoustic Phonetics (M.I.T. Press, Cambridge 1967)
- J.S. LIENARD Analyse, Synthèse et Reconnaissance Automatique de la Parole (Thèse d'Etat, Université de PARIS VI, 1972)
Les Processus de la Communication Parlée (Masson, Paris, 1977)
- J. LIFERMANN Théorie et Applications de la Transformation de Fourier Rapide (Masson, Paris, 1977)
- B. MALMBERG Les Domaines de la Phonétique (P.U.F., Paris, 1971)
(ed) Manual of Phonetics (North Holland, Amsterdam, 1968)
- J.J. MARIANI Contributions à la Reconnaissance Automatique de la Parole Utilisant la Notion de Spectre Différentiel (Thèse de Docteur Ingénieur, Université de PARIS VI, 1977)
- J.D. MARKEL & A.H. GRAY Linear Prediction of Speech (Springer-Verlag, Berlin, 1976)
- D.R. REDDY Speech Recognition by Machine : a Review (Proceedings of the I.E.E.E., N°64-4, 1976)
- D. TEIL Conception et Réalisation d'un Terminal à Réponse Vocale (Thèse de Docteur Ingénieur, Université de PARIS VI, 1975)
- J. VAISSIERE Contribution à la Synthèse par Règles du Français (Thèse de 3ème Cycle, Université de Grenoble, 1971)

LANGUE FRANCAISE

- LE NOUVEAU BESCHERELLE L'Art de Conjuguer (Hatier, Paris, 1966)
- J.C. CHEVALIER, Grammaire Larousse du Français Contemporain
- Cl. BLANCHE-BENVENISTE & al. (Larousse, Paris, 1964)
- P. DELATTRE Studies in French & Comparative Phonetics (Mouton, La Haye, 1966)

- P. FOUCHE Traité de Prononciation Française (Klincksieck, Paris, 1956)
- M. GREVISSE Le Bon Usage (Duculot, Gembloux, 1975)
- M. GROSS Méthodes en Syntaxe (Hermann, Paris, 1975)
- P.R. LEON Prononciation du Français Standard (Didier, Paris, 1966)
- B. MALMBERG La Phonétique Française (Hermods, Malmö, 1969)
- A. MARTINET Elements de Linguistique Générale (A. Colin, Paris, 1960)
- La Prononciation du Français Contemporain (Droz, Paris, 1945)
- A. RIGAULT (ed) La Grammaire du Français Parlé (Hachette, Paris, 1971)
- H. SEGUIN Le Genre des Adjectifs en Français, Langue Française N°20
(Larousse, Paris, 1973)



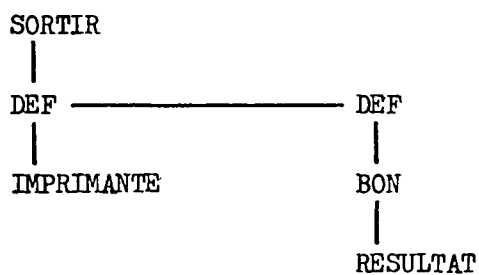


ENTREE SEMANTIQUE :

SORTIR (DEF (IMPRIMANTE), DEF (BON (RESULTAT)))



ARBORESCENCE SEMANTIQUE :



RESULTAT PHONETIQUE :

L<PRIM*T_S+R"LEB/R)ZULTA (lēprimāt//sər/ləborezylta)

"L'imprimante sort le bon résultat."

SOURCE LISTING

STMT LEV NT

```

1      0 PO:PROC OPTIONS(MAIN) ;
2      1 0 ON ERROR PUT DATA;
3      1 0 DCL NULL BUILTIN;
4      1 0 DCL LIMSI EXTERNAL ENTRY ;
5      1 0 CALL LIMSI ;
6      1 0 DCL AXEIN ENTRY (BIN FIXED(31), CHAR(80), BIN FIXED(31))
          OPTIONS (ASM);
7      1 0 DCL AXEDUT ENTRY (BIN FIXED(31), CHAR(80)) OPTIONS (ASM);
8      1 0 DCL CTLIN EXTERNAL ENTRY;
9      1 0 CALL CTLIN;

          /******/
          /*          */
          /* DICTIONNAIRE */
          /*          */
          /******/

          /****"LEX" ENTREE LEXICALE DU DICTIONNAIRE****/
          /****TOUT ELEMENT DE "LEX" SOUS-ENTEND "P->LEX"
          SI AUCUN POINTEUR N'EST SPECIFIE****/
10     1 0 DCL 1 LEX BASED (P),
          2 SEM CHAR(16) VAR, 2 CAT CHAR(1),
          2 MORPH CHAR(1), 2 REGL CHAR(1),
          2 PHON CHAR(12) VAR,
          2 PSUP PTR INIT (NULL),
          2 PFILS PTR INIT (NULL);
          /****FORMES SUPPLEMENTAIRES IRREGULIERES****/
11     1 0 DCL SUP (2) CHAR(12) VAR BASED; /****VERBES IRREGULIERS****/
12     1 0 DCL SUPIR (5) CHAR (12) VAR BASED; /****VERBES TRES IRREGULIERS****/

13     1 0 DCL DICO(26) PTR INIT ((26) NULL); /****DICTIONNAIRE****/
14     1 0 DCL ALPHABET (26) CHAR(1) INIT
          ('A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N',
           'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z' );

          /******/
          /****"POINTE" POINTE PAR "PT" A LA LETTRE DE RANG "R" SUR L'ENDROIT
          DANS LE DICTIONNAIRE OU DEVRAIT ETRE LE "SEME"****/
          /****"PT" EST NUL SI C'EST EN TETE DE LETTRE****/
          /******/

15     1 0 POINTE:PROC (SEME, PT, R);
16     2 0 DCL SEME CHAR(16) VAR, PT PTR, R FIXED;
17     2 0 DCL PX PTR;
18     2 0 DCL INIT CHAR(1);
19     2 0 INIT = SUBSTR (SEME, 1, 1); /****INITIALE****/

```

STMT LEV NT

```

20 2 0 R = 1;
21 2 0 DO WHILE (INIT /= ALPHABET(R) & R < 26);
22 2 1 R = R+1;
23 2 1 END;
    /**"R" INDIQUE DONC LE RANG DANS L'ALPHABET***/
24 2 0 PX = DICO(R);
25 2 0 PT = NULL; /**EN TETE***/
26 2 0 DO WHILE (PX /= NULL & SEM >= PX->SEM); /**SINON RECHERCHE***/
27 2 1 PT = PX;
28 2 1 PX = PX->LEX.PFILS;
29 2 1 END;
30 2 0 END POINTE;

    /*******
    /**"INSER" INSERE UNE ENTREE LEXICALE A L'ENDROIT INDIQUE PAR
    LE POINTEUR "PI" A LA LETTRE DE RANG "R"***/
    /**ON INSERE EN TETE DE LETTRE SI "PI" EST NUL***/
    /*******
31 1 0 INSER : PROC (PI, R);
32 2 0 DCL PI PTR, R FIXED;
33 2 0 IF PI = NULL THEN DO; /**EN TETE DE LISTE***/
34 2 1 LEX.PFILS = DICO(R);
35 2 1 DICO(R) = P;
36 2 1 END;
37 2 0 ELSE DO; /**AILLEURS QU'EN TETE***/
38 2 1 LEX.PFILS = PI->LEX.PFILS;
39 2 1 PI->LEX.PFILS = P;
40 2 1 END;
41 2 0 END INSER;

    /*******
    /**"CREDICO" REMPLIT LE DICTIONNAIRE DANS L'ORDRE ALPHABETIQUE
    AVEC LES ENTREES LEXICALES LUES EN DONNEES***/
    /*******
42 1 0 CREDICO :PROC;
43 2 0 DCL PR PTR;
44 2 0 DCL R FIXED;
45 2 0 PUT LIST ('DONNEES LEXICALES :'); PUT SKIP;
47 2 0 ALLOCATE LEX; SEM = '';
49 2 0 DO WHILE (SEM /= '1'); /**DERNIERE DONNEE : '1' ***/
50 2 1 ALLOCATE LEX SET (P);
    /**LECTURE***/
51 2 1 GET LIST (SEM, CAT, MORPH, REGL, PHON);
52 2 1 PUT SKIP;
53 2 1 PUT LIST (SEM, CAT, MORPH, REGL, PHON);
54 2 1 IF CAT = 'V' THEN DO;
55 2 2 IF MORPH = 'T' THEN DO; /**VERBES TRES IRREGULIERS***/
56 2 3 ALLOCATE SUPR SET (LEX.PSUP);
57 2 3 GET LIST (PSUP-> SUPR);

```

STMT LEV NT

```

58 2 3      PUT LIST (PSUP-> SUPIR);
59 2 3      END;
60 2 2      ELSE IF MORPH = 'I' THEN DO;      /**VERBES IRREGULIERS**/
61 2 3          ALLOCATE SUP SET (LEX.PSUP);
62 2 3          GET LIST (PSUP-> SUP);
63 2 3          PUT LIST (PSUP-> SUP);
64 2 3      END;
65 2 2      END;
66 2 1      CALL POINTE (SEM, PR, R);      /**RECHERCHE**/
67 2 1      CALL INSER (PR, R);      /**INSERTION**/
68 2 1      END;
69 2 0      END CREDICO;

      /*****
      /**"MEDICO" IMPRIME TOUT LE DICTIONNAIRE**/
      *****/

70 1 0      EDICO:PROC;
71 2 0          PUT SKIP (5);
72 2 0      PUT LIST ('DICTIONNAIRE CONSTRUIT :'); PUT SKIP;
74 2 0      DO J = 1 TO 26;      /**PARCOURS DES INITIALES**/
75 2 1          PUT SKIP;
76 2 1          PUT LIST ((10)' ');
77 2 1          PUT LIST (ALPHABET(J));
78 2 1          P = DICO(J);
79 2 1          DO WHILE (P /= NULL);      /**PARCOURS D'UNE SECTION**/
80 2 2              PUT SKIP;
81 2 2              PUT LIST ((10)' ');
82 2 2              PUT LIST (SEM, CAT, MORPH, REGL, PHON);
83 2 2              IF CAT = 'V' THEN DO;      /**VERBES IRREGULIERS**/
84 2 3                  IF MORPH = 'T' THEN PUT LIST (PSUP-> SUPIR);
85 2 3                  ELSE IF MORPH = 'I' THEN PUT LIST (PSUP-> SUP);
86 2 3              END;
87 2 2              P = LEX.PFELS;
88 2 2          END;
89 2 1      END;
90 2 0      END EDICO;

      /*****
      /**"CHERCHE" POINTE SUR LE "SEME" A CHERCHER DANS LE DICO,
      LE RESULTAT EST NUL SI LE "SEME" N'EST PAS TROUVE**/
      *****/

91 1 0      CHERCHE :PROC (SEME) RETURNS (PTR);
92 2 0      DCL SEME CHAR(16) VAR, PT PTR, R FIXED;
93 2 0      CALL POINTE (SEME, PT, R);
94 2 0      IF PT = NULL THEN RETURN (PT);      /**PAS DE MOT A CETTE LETTRE**/
95 2 0      IF PT->SEM /= SEME THEN PT = NULL;      /**SI PAS TROUVE**/
96 2 0      RETURN (PT);
97 2 0      END CHERCHE;

```

STMT LEV NT

```

/******/
/******/
/*          ARBORESCENCE SEMANTIQUE          */
/*          */
/******/

98  1  0  /***"ATOM" NOEUD DE L'ARBORESCENCE***/
      DCL 1 ATOM BASED (PA),
          2 MOT CHAR (16) VAR INIT(''),
          2 PDICO PTR INIT (NULL),
          2 ( PFILS, PFRER, PPER ) PTR INIT (NULL);

/******/
/*****"CREARBRE" LIT L'EXPRESSION EN ENTREE, CREE L'ARBORESCENCE
CORRESPONDANTE, ET EN RESULTAT POINTE SUR LA RACINE***/
/******/
99  1  0  CREAMBRE : PROC RETURNS (PTR);

/****SORTIE SUR CONSOLE****/
100  2  0  CALL AXEDUT (20, 'ENTREE_SEMANTIQUE:');
101  2  0  DCL NC BIN FIXED(31);
102  2  0  DCL LIN CHAR(80);
103  2  0  NC = 80;
/****ENTREE PAR CONSOLE****/
104  2  0  CALL AXEIN (NC, LIN, 0);
/****TAPER "#" A LA CONSOLE POUR TERMINER DIALOGUE****/
105  2  0  IF SUBSTR (LIN, 1, 1) = '#' THEN SIGNAL ENDFILE;
106  2  0  SUBSTR (LIN, NC+1, 1) = '#'; /****POUR MARQUER FIN DE CHAINE****/
107  2  0  DCL JC BIN FIXED INIT (0);

/****"CARSUIV" REVVOIE LE CARACTERE SUIVANT DE LA CHAINE****/
108  2  0  CARSUIV : PROC RETURNS (CHAR(1));
109  3  0  DCL CARS CHAR(1);
110  3  0  JC = JC + 1;
111  3  0  CARS = SUBSTR (LIN, JC, 1);
112  3  0  PUT EDIT (CARS)(A(1));
113  3  0  RETURN (CARS);
114  3  0  END CARSUIV;

115  2  0  DCL CAR CHAR (1) INIT (' ');
116  2  0  DCL (PACTU, PSUI, PTET) PTR;
117  2  0  DCL CONT REAL BINARY FIXED INIT (1);
118  2  0  PUT SKIP (5);
119  2  0  PUT LIST ('ENTREE SEMANTIQUE :');
120  2  0  PUT SKIP LIST ((19) '~');

```

STMT LEV NT

```

121  2  0      PUT SKIP (2);
          /**** CREATION DE LA RACINE ****/
122  2  0      ALLOCATE ATOM SET (PTET);  PACTU = PTET;
          /**** CREATION DES NOEUDS SUIVANTS ****/
          /****LES CARACTERES SONT LUS UN PAR UN****/
124  2  0      DO WHILE ( CAR ^= '#' );  /****DERNIER CARACTERE : '#' ****/
125  2  1          CAR = CARSUIV;  /****SUIVANT****/
126  2  1          DO WHILE ( CAR= ' ');  /*****ON IGNORE LES BLANCS*****/
127  2  2              CAR = CARSUIV;  /****SUIVANT****/
128  2  2          END;
129  2  1          DO WHILE ( 'A' <= CAR & CAR <= 'Z');  /****LECTURE DU MOT****/
130  2  2              PACTU->MOT= PACTU-> MOT || CAR;
131  2  2              CAR = CARSUIV;  /****SUIVANT****/
132  2  2          END;
133  2  1          IF CAR =',' THEN DO;  /*****CHAINAGE FILS*****/
134  2  2              ALLOCATE ATOM SET (PSUI);
135  2  2              CONT = CONT + 1;
136  2  2              PACTU->ATOM.PFILS= PSUI;
137  2  2              PSUI->PPER= PACTU;
138  2  2              PACTU= PSUI;
139  2  2          END;
140  2  1          IF CAR= ',' THEN DO;  /*****CHAINAGE FRERE*****/
141  2  2              ALLOCATE ATOM SET (PSUI);
142  2  2              CONT = CONT + 1;
143  2  2              PACTU->PPER= PSUI;
144  2  2              PSUI->PPER= PACTU->PPER;
145  2  2              PACTU= PSUI;
146  2  2          END;
147  2  1          IF CAR =')' THEN PACTU= PACTU->PPER;  /****REMONTER AU PERE****/
148  2  1      END;
149  2  0      CALL LIRARBRE (PTET, CONT);
150  2  0      RETURN (PTET);  /****RACINE****/
151  2  0      END CREARBRE;

          /*****
          /****"LIRARBRE" PARCOURT L'ARBORESCENCE REPEREE PAR "PTET"
          ET DE NOMBRE DE NOEUDS "CONT", EN IMPRIMANT SA DESCRIPTION
          QUI DOIT REDONNER L'EXPRESSION D'ORIGINE SI BIEN FORMEE****/
          /*****

152  1  0      LIRARBRE : PROC (PTET, CONT);
153  2  0          UCL (PACTU, PTET) PTR;
154  2  0          DCL CONT REAL BINARY FIXED;
155  2  0          PACTU = PTET;  /**** RACINE ****/
156  2  0          PUT SKIP (3);
157  2  0          PUT SKIP(2) LIST (' ARBORESCENCE CONSTRUITE : ');
158  2  0          PUT SKIP(2);
159  2  0          DO WHILE ( CONT >=1 );  /****PARCOURS DE L'ARBORESCENCE****/
160  2  1              PUT EDIT (PACTU->MOT) (A(16));  CONT =CONT -1;
162  2  1              DO WHILE ( PACTU->ATOM.PFILS ^= NULL);  /****AU FILS****/

```

STMT LEV NT

```

163 2 2      PACTU= PACTU->ATOM.PFILS;
164 2 2      PUT EDIT (' ( ' ) (A(3)));
165 2 2      PUT EDIT (PACTU->MOT) (A(16));  CONT =CONT -1;
167 2 2      END;
168 2 1      DO WHILE ( PACTU->PFRER=NULL & PACTU ~PTET );  /**AU PERE**/
169 2 2      PACTU= PACTU->PPER;
170 2 2      PUT EDIT (' ) ' ) (A(3)));
171 2 2      END;
172 2 1      IF PACTU ~PTET THEN DO;  /**ALLER AU FRERE**/
173 2 2      PACTU= PACTU->PFRER;
174 2 2      PUT EDIT (' , ' ) (A(3)));
175 2 2      END;
176 2 1      END;
177 2 0      IF PACTU ~PTET THEN PUT LIST ('  EXPRESSION MAL FORMEE');
          /****CAR ON N'EST PAS REVENU A LA RACINE***/
178 2 0      END LIRARBRE;

          /*****/

          /*****/
          /*      MORPHOLOGIE ET SYNTAXE      */
          /*****/

          /****GNP" SPECIFIE GENRE , NOMBRE , ET PERSONNE***/
          /****ET INDIQUE S'IL Y A UN PRONOM***/
179 1 0      DCL 1 GNP BASED (PG),  2 PFILS PTR INIT (NULL),
          2 FEM BIN FIXED(1) INIT(0),
          2 PLU BIN FIXED(1) INIT(0),
          2 PERS DEC FIXED(1) INIT(3),
          2 PRON BIN FIXED(1) INIT(0);

          /*****/
          /****"DETERGNP" PARCOURT L'ARBORESCENCE REPEREE PAR "PTETA",
          CONSULTE LE DICO A CHAQUE NOEUD, ET DETERMINE AINSI LA LISTE
          DES GENRES, NOMBRES, ET PERSONNES, DONT ON RENVOIE LA TETE***/
          /*****/
180 1 0      DETERGNP : PROC (PTETA) RETURNS (PTR);
181 2 0      DCL (PTETA, PTETG) PTR;
182 2 0      DCL PNOUV PTR;
183 2 0      DCL PP PTR;
184 2 0      P = PTETA;  /****RACINE DE L'ARBORESCENCE***/
185 2 0      ALLOCATE GNP SET (PTETG);  /****TETE DE LISTE***/
186 2 0      PG = PTETG;
187 2 0      PUT SKIP (3) LIST ('APPELS AU DICO :');

```

STMT LEV NT

```

188 2 0 PUT SKIP (2);
189 2 0 DCL TET BIN FIXED(1) INIT(1);
190 2 0 DO WHILE (P /= PTETA | TET = 1);  /****PARCOURS DE L'ARBRE****/
191 2 1 IF TET = 1 THEN TET = 0;
192 2 1 PP = P;
193 2 1 DO WHILE (PP /= NULL);  /****AU FILS****/
194 2 2 PP->ATOM.PDICO = CHERCHE (PP->ATOM.MOT);  /****APPEL AU DICO****/
195 2 2 IF PP->ATOM.PDICO = NULL THEN DO;  /****SI PAS TROUVE****/
196 2 3 PUT LIST (PP->ATOM.MOT);
197 2 3 PUT LIST ('(->ABSENT DU DICO)');
198 2 3 END;
199 2 2 ELSE DO;  /****SI BIEN TROUVE****/
200 2 3 PUT LIST (PP->ATOM.PDICO -> LEX.PHON);
/*****TRAITEMENT DU GENRE*****/
201 2 3 DCL PQ PTR; PQ = PP->ATOM.PDICO;
203 2 3 IF PQ->LEX.CAT = 'N' THEN  /****NOM****/
IF PQ->LEX.REGL = 'F' THEN PG->FEM = 1;  /****GENRE****/
204 2 3 END;
/*****TRAITEMENT DU NOMBRE*****/
205 2 2 IF PP->ATOM.MOT = 'PLU' THEN DO;
206 2 3 PG->PLU = 1;  /****NOMBRE****/
207 2 3 PG->PERS = 6;  /****PERSONNE****/
208 2 3 END;
/*****TRAITEMENT DE LA PERSONNE*****/
209 2 2 ELSE DO;
210 2 3 IF PP->ATOM.MOT = 'MOI' THEN PG->PERS = 1;
211 2 3 IF PP->ATOM.MOT = 'TOI' THEN PG->PERS = 2;
212 2 3 IF PP->ATOM.MOT = 'VOUS' THEN PG->PERS = 4;
213 2 3 IF PP->ATOM.MOT = 'VOUS' THEN PG->PERS = 5;
/*****TRAITEMENT DU PRONOM*****/
214 2 3 IF PERS /= 3 & PERS < 6 THEN PG->PRON = 1;  /****1,2,4,5****/
215 2 3 IF PP->ATOM.MOT = 'LUI' THEN PG->PRON = 1;
216 2 3 IF PP->ATOM.MOT = 'ELLE' THEN DO;
217 2 4 PG->FEM = 1; PG->PRON = 1;
219 2 4 END;
220 2 3 END;
221 2 2 P = PP; PP = PP->ATOM.PFILS;  /****FILS SUIVANT****/
223 2 2 END;  /****FIN DE BOUCLE AU FILS****/
224 2 1 DO WHILE (P->ATOM.PFRER = NULL & P->ATOM.PPER /= NULL);
225 2 2 P = P->ATOM.PPER;  /****AU PERE****/
226 2 2 END;
227 2 1 IF P /= PTETA THEN DO;
228 2 2 P = P->ATOM.PFRER;  /****AU FRERE****/
229 2 2 ALLOCATE GNP SET (PNOUV);  /****SUITE DE LA LISTE****/
230 2 2 PG->GNP.PFILS = PNOUV;
231 2 2 PG = PNOUV;
232 2 2 END;
233 2 1 END;  /****FIN DE BOUCLE DE L'ARBRE****/
234 2 0 RETURN (PTETG);  /****TETE DE LISTE DES "GNP"****/

```

STMT LEV NT

```

235 2 0 END DETERGNP;

      /*****
      ****"ACCORVRB" CONJUGUE AU PRESENT ET RENVOIE LE VERBE
      REPERE DANS LE DICO PAR LE POINTEUR "PD"****/
      *****/
236 1 0 ACCORVRB : PROC (PD) RETURNS (CHAR(16) VAR);
237 2 0 DCL PD PTR, FORM CHAR(16) VAR;
238 2 0 P = PD; /****"LEX" SOUS-ENTEND DONC "PD->LEX"****/
      /****"PERS" SOUS-ENTEND "PG->PERS"****/
      /****FORMES IRREGULIERES SJNT DANS LE DICO****/
239 2 0 IF LEX.MORPH = 'T' THEN DO; /****TRES IRREGULIERS****/
240 2 1 IF PERS = 1 THEN FORM = LEX.PHON; /****1****/
241 2 1 ELSE FORM = LEX.PSUP->SUP (PERS-1); /****2,3,4,5,6****/
242 2 1 END;
243 2 0 ELSE DO;
244 2 1 IF LEX.MORPH = 'I' THEN DO; /****IRREGULIERS****/
245 2 2 IF PERS = 6 THEN FORM = LEX.PSUP->SUP (2); /****6****/
246 2 2 ELSE DO;
247 2 3 IF PERS > 3 THEN FORM = LEX.PSUP->SUP (1); /****4,5****/
248 2 3 ELSE FORM = LEX.PHON; /****1,2,3****/
249 2 3 END;
250 2 2 END;
251 2 1 ELSE DO;
252 2 2 FORM = LEX.PHON; /****REGULIERS****/
      /****REGULIERS DU 3EME GROUPE****/
253 2 2 IF LEX.MORPH = '3' & PERS < 4 THEN
      /****SUPPRESSION DE LA CONSONNE FINALE****/
      FORM = SUBSTR (FORM, 1, LENGTH(FORM) -1); /****1,2,3****/
      /****REGULIERS DU 2EME GROUPE****/
254 2 2 ELSE IF LEX.MORPH = '2' THEN DO;
255 2 3 FORM = FORM || 'I';
256 2 3 IF PERS > 3 THEN FORM = FORM || 'S'; /****4,5,6****/
257 2 3 END;
258 2 2 END;
259 2 1 IF PERS = 4 THEN FORM = FORM || '/'; /****ONS****/
260 2 1 ELSE IF PERS = 5 THEN FORM = FORM || 'E'; /****EZ****/
261 2 1 END;

262 2 0 IF LEX.REGL = 'P' THEN DO; /****PRONOMINAUX****/
263 2 1 DCL REFL (6) CHAR(4) VAR INIT
      ('ME,', 'TE,', 'SE,', 'NWZ.', 'VWZ.', 'SE,');
264 2 1 FORM = REFL (PERS) || FORM;
265 2 1 END;

266 2 0 RETURN (FORM); /****VERBE CONJUGUE****/
267 2 0 END ACCORVRB;

```

```

      /****

```

STMT LEV NT

```

      /****"NASAL" NASALISE ET RENVOIE LA VOYELLE "VOC"****/
      /******/
268   1  0  NASAL : PROC (VOC) RETURNS (CHAR(1));
269   2  0  DCL VOC CHAR(1);
270   2  0  IF VOC = '(' | VOC = 'I' THEN VOC = '<';  /****"IN"****/
271   2  0  ELSE IF VOC = 'A' THEN VOC = '*';  /****"AN"****/
272   2  0  IF VOC = '+' THEN VOC = '/';  /****"ON"****/
273   2  0  RETURN (VOC);  /****VOYELLE NASALISEE****/
274   2  0  END NASAL;

      /******/
      /****"BEL" MODIFIE ET RENVOIE LA FINALE "FINADJ"****/
      /******/
275   1  0  BEL : PROC (FINADJ) RETURNS (CHAR(2));
276   2  0  DCL FINADJ CHAR(2);
277   2  0  IF FINADJ = 'L' THEN FINADJ = 'O ';  /****BEL -> BEAU****/
278   2  0  ELSE IF FINADJ = 'L' THEN FINADJ = 'W ';  /****MOL -> MOU****/
279   2  0  IF FINADJ = 'Y' THEN FINADJ = '= ';  /****VIEIL -> VIEUX****/
280   2  0  RETURN (FINADJ);  /****FINALE MODIFIEE****/
281   2  0  END BEL;

      /******/
      /****"ACCORADJ" ACCORDE ET RENVOIE L'ADJECTIF
      REPERE DANS LE DICO PAR LE POINTEUR "PD"****/
      /******/
282   1  0  ACCORADJ : PROC (PD) RETURNS (CHAR(12) VAR);
283   2  0  DCL PD PTR, ADJ CHAR(12) VAR;
284   2  0  ADJ = PD->LEX.PHON;
285   2  0  DCL MORF CHAR(1);
286   2  0  MORF = PD->LEX.MORPH;
287   2  0  DCL ANT BIN FIXED (1) INIT (0);
288   2  0  IF PD->LEX.REGL = 'A' THEN ANT = 1;
289   2  0  DCL LG BIN FIXED;
290   2  0  LG = LENGTH (ADJ);
291   2  0  IF PD->FEM = 0 THEN DO;  /****SI MASCULIN MODIFIER FINALE****/
292   2  1  IF MORF = 'S' THEN DO;  /****SUBSTITUTION****/
293   2  2  IF SUBSTR (ADJ, LG) = 'V' THEN SUBSTR (ADJ, LG) = 'F';
294   2  2  ELSE DO;
295   2  3  IF SUBSTR (ADJ, LG-1) = '=Z' THEN SUBSTR (ADJ, LG-1) = 'aR';
296   2  3  ELSE DO;
297   2  4  IF SUBSTR (ADJ, LG-3) = 'TRIS' THEN SUBSTR (ADJ, LG-2) = 'aR ';
298   2  4  ELSE IF ADJ = 'S(X' THEN ADJ = 'S(K';  /****SECHE -> SEC****/
299   2  4  END;
300   2  3  END;
301   2  2  END;  /****FIN DE SUBSTITUTION****/
302   2  1  ELSE DO;
303   2  2  IF MORF = 'A' THEN DO;  /****DES-ADJONCTION****/
      /****FERMER VOYELLE FINALE SAUF DEVANT "N"****/
304   2  3  IF SUBSTR (ADJ, LG) = 'N' THEN DO;

```

STMT LEV NT

```

305 2 4   IF SUBSTR (ADJ, LG-1, 1) = '(' THEN
          SUBSTR (ADJ, LG-1, 1) = ')';
306 2 4   ELSE IF SUBSTR (ADJ, LG-1, 1) = '+' THEN
          SUBSTR (ADJ, LG-1, 1) = '0';
307 2 4   END;
          /**DEVANT "N" NASALISER LA VOYELLE "U"****/
308 2 3   ELSE IF SUBSTR (ADJ, LG-1, 1) = 'U' THEN
          SUBSTR (ADJ, LG-1, 1) = '<';
          /**SI ANTERIEUR ET SINGULIER, MARQUER LIAISON POSSIBLE***/
309 2 3   IF ANT = 1 & PG->PLU = 0 THEN ADJ = ADJ || '.'; /*LIAISON*/
310 2 3   ELSE DO; /***SI POSTERIEUR OU PLURIEL***/
          /**NASALISER VOYELLE FINALE DEVANT "N"****/
311 2 4   IF SUBSTR (ADJ, LG) = 'N' THEN
          SUBSTR (ADJ, LG-1, 1) = NASAL (SUBSTR (ADJ, LG-1, 1));
312 2 4   ADJ = SUBSTR (ADJ, 1, LG-1); /**ENLEVER CONSONNE FINALE***/
313 2 4   END;
314 2 3   END; /**FIN DE DES-ADJONCTION***/
315 2 2   ELSE DO;
316 2 3   IF MORF = 'I' THEN DO; /**IRREGULIER TYPE "BEL"****/
          /**SI ANTERIEUR ET SINGULIER, AJOUTER MARQUEUR SPECIAL***/
317 2 4   IF ANT = 1 & PG->PLU = 0 THEN ADJ = ADJ || ':'; /*LIAISON*/
          /**SINON MODIFIER FINALE MAINTENANT***/
318 2 4   ELSE SUBSTR (ADJ, LG-1) = BEL (SUBSTR (ADJ, LG-1));
319 2 4   END; /**FIN DU TYPE "BEL"****/
320 2 3   END;
321 2 2   END;
322 2 1   END; /**FIN DU MASCULIN***/
323 2 0   IF PG->PLU = 1 THEN DO; /**SI PLURIEL***/
324 2 1   IF MORF = 'O' & PG->FEM = 0 THEN /**PLURIEL MASCULIN EN "O"****/
          IF SUBSTR (ADJ, LG-1) = 'AL' THEN SUBSTR (ADJ, LG-1) = 'O ';
325 2 1   IF ANT = 1 THEN ADJ = ADJ || 'Z.'; /**PLURIEL ET LIAISON***/
326 2 1   END; /**FIN DU PLURIEL***/
327 2 0   RETURN (ADJ); /**ADJECTIF ACCORDE***/
328 2 0   END ACCRADJ;

          /*******
          /**"ACCORNOM" ACCORDE ET RENVOIE LE NOM
          REPERE DANS LE DICO PAR LE POINTEUR "PD"****/
          /*******
329 1 0   ACCORNOM : PROC (PD) RETURNS (CHAR(12) VAR);
330 2 0   DCL PD PTR, NOM CHAR(12) VAR;
331 2 0   NOM = PD->LEX.PHON;
332 2 0   IF PG->PLU = 1 THEN DO; /**MISE AU PLURIEL***/
333 2 1   IF PD->LEX.MORPH = 'O' THEN DO;
334 2 2   DCL LG BIN FIXED; LG = LENGTH (NOM);
336 2 2   SUBSTR (NOM, LG-1, 2) = 'O '; /**PLURIEL EN 'O'****/
337 2 2   END;
338 2 1   ELSE IF NOM = 'ay' THEN NOM = 'Y'; /**"OEIL" -> "YEUX"****/
339 2 1   END;

```

STMT LEV NT

```

340 2 0 RETURN (NOM);  /***NOM ACCORDE***/
341 2 0 END ACCORNOM;

/*****/

/****"SON" ELEMENT PHONETIQUE DE LA PHRASE DE SURFACE***/
342 1 0 DCL 1 SON BASED (PS),
      2 FORM CHAR(20) VAR INIT (''),
      2 PFILS PTR INIT (NULL);
/****"SON" SOUS-ENTEND "PS->SON" SI NON AUTREMENT SPECIFIE***/

/*****/
/****"SYNTAX" PARCOURT L'ARBORESCENCE REPEREE PAR "PTETA"
ET SELON LA LISTE D'ACCORD REPEREE PAR "PTETG",
CONSTRUIT ET RENVOIE LA PHRASE PHONETIQUE DE SURFACE***/
/*****/
343 1 0 SYNTAX : PROC (PTETA,PTETG) RETURNS (CHAR(100) VAR);
344 2 0 DCL (PTETA,PTETG) PTR;
345 2 0 DCL PHRASE CHAR(100) VAR INIT ('');
346 2 0 DCL PTETS PTR;
347 2 0 DCL PLZ PTR;

/****"PN" POITE SUR LE DERNIER "SON" DE LA LISTE
ET "PRC" SUR L'AVANT-DERNIER "SON"***/
348 2 0 DCL PN PTR; DCL PRC PTR;
350 2 0 PA = PTETA;  /****"ATOM" SOUS-ENTEND "PA->ATOM"***/
351 2 0 PG = PTETG;  /****"GNP" SOUS-ENTEND "PG->GNP"***/
352 2 0 ALLOCATE SON SET (PTETS);  /****TETE DE LISTE***/
353 2 0 PN = PTETS;
354 2 0 DCL QU BIN FIXED(1) INIT (0);
355 2 0 DCL NEG BIN FIXED(1) INIT(0);
356 2 0 DCL A BIN FIXED(1) INIT(0);
357 2 0 DCL D BIN FIXED(1) INIT(0);
358 2 0 DCL DE BIN FIXED(1);
359 2 0 DCL INVERT DEC FIXED(1);
360 2 0 DCL COL DEC FIXED INIT(0);
361 2 0 DO WHILE (PA /= PTETA | COL = 0);  /****PARCOURS DE L'ARBRE***/
362 2 1 COL = COL + 1;  /****COMPTE LES COLONNES***/
363 2 1 IF D = 1 & COL = 2 THEN DE = 1; ELSE DE = 0;  /*PARTICULE "DE"*/
365 2 1 INVERT = 0;  /****INVERSION***/
366 2 1 DO WHILE (PA /= NULL);  /****AU FILS***/
367 2 2 ALLOCATE SON SET (PS);
      /****"." INDIQUE LIAISON, ET "," ELISION POSSIBLES***/
      IF ATOM.MOT = 'QU' THEN DO;  /****QUESTION***/
368 2 2 SON.FORM = '(SKE,'; QU = 1;
369 2 3 END;
371 2 3 ELSE IF ATOM.MOT = 'NEG' THEN NEG = 1;  /****NEGATION***/
372 2 2 IF ATOM.MOT = 'DEF' THEN DO;  /****ARTICLE DEFINI***/
373 2 2 IF PLU = 1 THEN SON.FORM = 'L)Z.';  /****LES***/
374 2 3 ELSE DO;
375 2 3

```

STMT LEV NT

```

376 2 4      IF FEM = 1 THEN SON.FORM = 'LA,'; /* "," POUR ELISION*/
377 2 4      ELSE SON.FORM = 'LE,';
378 2 4      END;
379 2 3      END; /****FIN DU DEFINI****/
380 2 2      ELSE IF ATOM.MOT = 'INO' THEN DO; /****ARTICLE INDEFINI****/
381 2 3      IF NEG = 1 & COL = 2 & DE = 0 & A = 0 THEN SON.FORM = 'DE,';
382 2 3      ELSE DO;
383 2 4          IF PLU = 1 & DE = 0 THEN SON.FORM = 'DIZ,'; /****DES****/
384 2 4          ELSE DO;
385 2 5              IF PLU = 0 THEN DO;
386 2 6                  IF FEM = 1 THEN SON.FORM = 'UN,'; /****UNE****/
387 2 6                  ELSE SON.FORM = 'N,'; /****UN****/
388 2 6              END;
389 2 5          END;
390 2 4      END;
391 2 3      END; /****FIN DE L'INDEFINI****/
392 2 2      IF ATOM.MOT = 'PART' THEN DO; /****ARTICLE PARTITIF****/
393 2 3      IF DE = 0 THEN DO;
394 2 4          IF NEG = 1 & COL = 2 & A = 0 THEN SON.FORM = 'DE,';
395 2 4          ELSE DO;
396 2 5              IF FEM = 1 THEN SON.FORM = 'DE LA,';
397 2 5              ELSE SON.FORM = ';DE,LE,';
398 2 5              /****";" INDIQUE CONTRACTION POSSIBLE****/
399 2 4          END;
400 2 3      END;
401 2 2      END; /****FIN DU PARTITIF****/
402 2 2      IF ATOM.PDICO->LEX.CAT = 'N' THEN SON.FORM = ACCORNOM (PDICO);
403 2 2      IF ATOM.PDICO->LEX.CAT = 'A' THEN SON.FORM = ACCORADJ (PDICO);
404 2 2      IF ATOM.PDICO->LEX.CAT = 'V' THEN DO; /****VERBE****/
405 2 3          INVERT = 1; /****DECLENCHERA L'INVERSION****/
406 2 3          SON.FORM = ACCORVRB (PDICO);
407 2 3          /****NEGATION****/
408 2 4          IF NEG = 1 THEN SON.FORM = 'NE,' || SON.FORM || 'PA';
409 2 4          IF PRON = 1 THEN DO; /****PRONOM SUJET****/
410 2 4              DCL PROS CHAR(4) VAR;
411 2 4              IF PERS = 1 THEN PROS = 'JE,';
412 2 4              IF PERS = 2 THEN PROS = 'TU,';
413 2 4              IF PERS = 4 THEN PROS = 'NWZ,'; /*NOUS*/
414 2 4              IF PERS = 5 THEN PROS = 'VWZ,'; /*VOUS*/
415 2 5              IF PERS = 3 THEN DO;
416 2 5                  IF FEM = 1 THEN PROS = '(L,'; /*ELLE*/
417 2 5                  ELSE PROS = 'IL,';
418 2 5              END;
419 2 5              ELSE IF PERS = 6 THEN DO;
420 2 5                  IF FEM = 1 THEN PROS = '(LZ,'; /*ELLES*/
421 2 5                  ELSE PROS = 'ILZ,'; /*ILS*/
422 2 4              END;
423 2 4              SON.FORM = PROS || SON.FORM;
424 2 4          END;

```

STMT LEV NT

```

423 2 3      SON.FORM = '_' || SON.FORM || '$';  /*MARQUEURS SYNTAXIQUES*/
          /****;" INDIQUE UNE CONTRACTION POSSIBLE****/
424 2 3      IF ATOM.PDICO->LEX.REGL = 'A' THEN DO;  /****A****/
425 2 4          SON.FORM = SON.FORM || 'A';  A = 1;
427 2 4      END;
428 2 3      ELSE IF ATOM.PDICO->LEX.REGL = 'D' THEN DO;  /****DE****/
429 2 4          SON.FORM = SON.FORM || 'DE, ' ;  D = 1;
431 2 4      END;
432 2 3      END;  /****FIN DU GROUPE VERBAL****/
          /****POSITIONNEMENT DU NOUVEAU MOT EN SURFACE****/
433 2 2      IF INVERT = 1 & PDICO->CAT ^= 'V' THEN DO;  /****INSERER****/
434 2 3          PS->SON.PFILS = PRC->SON.PFILS;
435 2 3          PRC->SON.PFILS = PS;  PRC = PS;
437 2 3      END;
438 2 2      ELSE DO;  /****AJOUTER A LA SUITE****/
439 2 3          PN->SON.PFILS = PS;
440 2 3          PRC = PN;  PN = PS;
442 2 3      END;
          /****ALLER AU FILS SUIVANT****/
443 2 2      PZ = PA;  /****SAUVE PA****/
444 2 2      PA = PA->ATOM.PFILS;
445 2 2      END;  /****FIN DE BOUCLE AU FILS****/
446 2 1      PA = PZ;  /****RESTAURE PA****/
447 2 1      DO WHILE (ATOM.PFRER = NULL & ATOM.PPER ^= NULL);
448 2 2          PA = PA->ATOM.PPER;  /****AU PERE****/
449 2 2      END;
450 2 1      IF PA ^= PTETA THEN DO;
451 2 2          PA = PA->ATOM.PFRER;  /****AU FRERE****/
452 2 2          PG = PG->GNP.PFILS;  /****"GNP" SUIVANT****/
453 2 2      END;
454 2 1      END;  /****FIN DE LA BOUCLE DE L'ARBRE****/
455 2 0      PUT SKIP (3) LIST ('MORPHOLOGIE ET SYNTAXE : ');
456 2 0      PUT SKIP (2);
          /****CONCATENATION DE LA LISTE DES "SON"****/
457 2 0      PS = PTETS;
458 2 0      DO WHILE (PS ^= NULL);
459 2 1          PUT LIST (SON.FORM);
460 2 1          PHRASE = PHRASE || SON.FORM;  /****CONCATENATION****/
461 2 1          PS = SON.PFILS;
462 2 1      END;
463 2 0      IF QU = 1 THEN PHRASE = PHRASE || '?';  /****MARQUEUR****/
464 2 0      PUT SKIP (3) DATA (PHRASE);
465 2 0      RETURN (PHRASE);  /****PHRASE PHONETIQUE DE SURFACE****/
466 2 0      END SYNTAX;

          /******
          /****"CONS" INDIQUE SI "CARC" EST UNE CONSONNE OU UN MARQUEUR***
          /******
467 1 0      CONS : PROC (CARC) RETURNS (BIN FIXED (1));

```

STMT LEV NT

```

468 2 0 DCL CARC CHAR(1);
469 2 0 DCL CNS BIN FIXED (1) INIT (0);
      /***CONSONNES ET MARQUEURS SYNTAXIQUES***/
470 2 0 DCL CONSONNES (22) CHAR(1) INIT
      ('P', 'B', 'T', 'D', 'K', 'G', 'F', 'V', 'S', 'Z', 'X', 'J',
       'M', 'N', 'L', 'R', 'H', 'I', 'A', 'E', 'O', 'U');
471 2 0 DO K= 1 TO 22;
472 2 1   IF CARC = CONSONNES (K) THEN DO;
473 2 2     CNS = 1;
474 2 2     GOTO FIN;
475 2 2   END;
476 2 1 END;
477 2 0   FIN :
      RETURN (CNS);   /***CONSONNE OU NON***/
478 2 0 END CONS;

      /***LIAISON EFFECTUEE LIAISONS ET ELISIONS SUR LA CHAINE "FRAZ"***/
      /***LIAISON : PROC (FRAZ);***/
479 1 0 LIAISON : PROC (FRAZ);
480 2 0 DCL FRAZ CHAR(100) VAR;
481 2 0 DCL CAR CHAR(1);
482 2 0 UCL L BIN FIXED;
483 2 0 FRAZ = FRAZ || '#';   /***MARQUEUR FINAL***/
484 2 0 L = LENGTH (FRAZ);
485 2 0 DO I= 1 TO L-1;   /***PARCOURS DE LA CHAINE***/
486 2 1   CAR = SUBSTR (FRAZ, I, 1);   /***CARACTERE SUIVANT***/
487 2 1   IF CAR = ';' THEN DO;   /***CONTRACTION EVENTUELLE***/
488 2 2     SUBSTR (FRAZ, I, 1) = ' ';   /***EFFACER LE ";"***/
489 2 2     IF SUBSTR (FRAZ, I+1, 1) = 'A' THEN DO;
         /***"A LE" DEVIENT "AU" DEVANT CONSONNE***/
490 2 3       IF SUBSTR (FRAZ, I+2, 3) = 'LE,' THEN DO;
491 2 4         IF CONS (SUBSTR (FRAZ, I+5, 1)) = 1 THEN
             SUBSTR (FRAZ, I+1, 4) = 'O  ';   /***AU***/
492 2 4       END;
         /***"A LES" DEVIENT "AUX"***/
493 2 3       ELSE IF SUBSTR (FRAZ, I+2, 4) = 'L)Z.' THEN
             SUBSTR (FRAZ, I+1, 3) = 'O  ';   /***AUX***/
494 2 3     END;
495 2 2   ELSE DO;   /***PARTICULE "DE"***/
         /***"DE LE" DEVIENT "DU" DEVANT CONSONNE***/
496 2 3       IF SUBSTR (FRAZ, I+4, 3) = 'LE,' THEN DO;
497 2 4         IF CONS (SUBSTR (FRAZ, I+7, 1)) = 1 THEN
             SUBSTR (FRAZ, I+2, 5) = 'U  ';   /***DU***/
498 2 4       END;
         /***"DE LES" DEVIENT "DES"***/
499 2 3       ELSE IF SUBSTR (FRAZ, I+4, 4) = 'L)Z.' THEN
             SUBSTR (FRAZ, I+2, 3) = '  ';   /***DES***/
500 2 3     END;

```

STMT LEV NT

```

501 2 2  END;  /**FIN DE CONTRACTION***/
502 2 1  IF CAR = '.' THEN DO;  /**LIAISON EVENTUELLE***/
503 2 2  SUBSTR (FRAZ, I, 1) = ' ';  /**EFFACER LE "."***/
      /**EFFACER CONSONNE DE LIAISON DEVANT CONSONNE OU MARQUEUR***/
504 2 2  IF CONS (SUBSTR (FRAZ, I+1, 1)) = 1 THEN DO;
      /**NASALISER VOYELLE DEVANT "N"***/
505 2 3  IF SUBSTR (FRAZ, I-1, 1) = 'N' THEN
      SUBSTR (FRAZ, I-2, 1) = NASAL (SUBSTR (FRAZ, I-2, 1));
      /**EFFACER CONSONNE DE LIAISON***/
506 2 3  SUBSTR (FRAZ, I-1, 1) = ' ';
507 2 3  END;
508 2 2  ELSE DO;  /**DEVANT VOYELLE, MODIFIER CONSONNE DE LIAISON***/
509 2 3  IF SUBSTR (FRAZ, I-1, 1) = 'S' THEN
      SUBSTR (FRAZ, I-1, 1) = 'Z';  /**S -> Z***/
510 2 3  ELSE IF SUBSTR (FRAZ, I-1, 1) = 'D' THEN
      SUBSTR (FRAZ, I-1, 1) = 'T';  /**D -> T***/
511 2 3  END;
512 2 2  END;  /**FIN DE LIAISON***/
513 2 1  ELSE DO;
514 2 2  IF CAR = ':' THEN DO;  /**LIAISON TYPE "BEL"***/
515 2 3  SUBSTR (FRAZ, I, 1) = ' ';  /**EFFACER MARQUEUR ":"***/
      /**MODIFIER FINALE DEVANT CONSONNE OU MARQUEUR***/
516 2 3  IF CONS (SUBSTR (FRAZ, I+1, 1)) = 1 THEN
      SUBSTR (FRAZ, I-2, 2) = BEL (SUBSTR (FRAZ, I-2, 2));
517 2 3  END;
518 2 2  ELSE IF CAR = ',' THEN DO;  /**ELISION EVENTUELLE***/
519 2 3  SUBSTR (FRAZ, I, 1) = ' ';  /**EFFACER LE ","***/
      /**EFFACER VOYELLE A ELIDER DEVANT VOYELLE***/
520 2 3  IF CONS (SUBSTR (FRAZ, I+1, 1)) = 0 THEN
      SUBSTR (FRAZ, I-1, 1) = ' ';
521 2 3  END;
522 2 2  END;
523 2 1  END;  /**FIN DE BOUCLE***/
524 2 0  FRAZ = SUBSTR (FRAZ, 1, L-1);  /**SUPPRESSION DU "#" FINAL***/
      /**ELIMINATION DES " " ET DES "H"***/
525 2 0  DCL AUX CHAR(130) VAR;
526 2 0  AUX = FRAZ;  FRAZ = ' ';  /**ON SAUVE "FRAZ"***/
528 2 0  DO I= 1 TO L-1;  /**PARCOURS DE LA CHAINE***/
529 2 1  CAR = SUBSTR (AUX, I, 1);  /**CARACTERE SUIVANT***/
530 2 1  IF CAR ^= ' ' & CAR ^= 'H' THEN FRAZ = FRAZ || CAR;
531 2 1  END;
532 2 0  END LIAISON;

```

```

/*****

```

```

/*****
/**"CONVICO" CONVERTIT CHAINE PL/I "STR" EN TABLEAU FORTRAN A1
ET L'ENVOIE A LA PROCEDURE FORTRAN POUR SYNTHESE PAR ICO 6***/

```

STMT LEV NT

```

/*****/
533 1 0 CONVICO : PROC (STR);
534 2 0 DCL STR CHAR(100) VAR;
535 2 0 DCL ICSICI ENTRY ((100) CHAR(2), BIN FIXED(31), BIN FIXED(31),
      BIN FIXED(31)) OPTIONS (FORTRAN);
536 2 0 DCL PHONE (100) CHAR(2);
537 2 0 PHONE = ' ';
538 2 0 DCL LS BIN FIXED(31); LS = LENGTH (STR);
540 2 0 DO IC = 1 TO LS; /***CONVERSION AU FORMAT FORTRAN A1***/
541 2 1   PHONE (IC) = SUBSTR (STR, IC, 1) || ' ';
542 2 1 END;
543 2 0 CALL ICSICI (PHONE, LS, 3, 1); /***PROCEDURE FORTRAN***/
544 2 0 END CONVICO;

/*****/

/****DICTIONNAIRE****/
545 1 0 CALL CREDICO;
546 1 0 CALL EDICO;

547 1 0 ON ENDFILE GOTO FINI;
548 1 0 DO IB =1 TO 50;
549 1 1   PUT PAGE;
/****ARBORESCENCE SEMANTIQUE****/
550 1 1 DCL PTETA PTR;
551 1 1 PTETA = CREARBRE;
/****GENRE, NOMBRE, ET PERSONNE****/
552 1 1 DCL PTETG PTR;
553 1 1 PTETG = DETERGNP (PTETA);
/****LECTURE DE TOUT LES "GNP"****/
554 1 1 PUT SKIP (3) LIST ('GENRE, NOMBRE, PERSONNE, PRONOM :');
555 1 1 PUT SKIP;
556 1 1 P = PTETG;
557 1 1 DO WHILE (P /= NULL);
558 1 2   PUT SKIP;
559 1 2   PUT LIST (P->GNP);
560 1 2   P = P->GNP.PFELS;
561 1 2 END;
/****PHRASE PHONETIQUE DE SURFACE****/
562 1 1 DCL PHRASE CHAR(100) VAR;
563 1 1 PHRASE = SYNTAX (PTETA, PTETG);
/****LIAISONS ET ELISIONS****/
564 1 1 CALL LIAISON (PHRASE);
/****EDITION DU RESULTAT FINAL****/
565 1 1 PUT SKIP (3);
566 1 1 PUT SKIP (5) LIST ('SORTIE PHONETIQUE :');
567 1 1 PUT LIST (PHRASE);
568 1 1 PUT SKIP LIST ((19)'-');

```

STMT LEV NT

```
569 1 1  /**SYNTHESE PAR ICPHONE***/  
      CALL CONVICO (PHRASE);  
570 1 1  END;  
571 1 0  FINI :  
      END PO;
```

DONNEES LEXICALES :

ENFANT	N	R	M	*F*
HOMME	N	R	M	+M
VIN	N	R	M	V<
SOUPE	N	R	F	SWP
CHEVAL	N	O	M	XEVAL
FEMME	N	R	F	FAM
EAU	N	R	F	O
PLUME	N	R	F	PLUM
TRAVAIL	N	O	M	TRAVAY
JEUNE	A	R	A	JØN
BON	A	A	A	B+N
JOLI	A	R	A	JOLI
BEAU	A	I	A	BIL
PETIT	A	A	A	PETIT
GRAND	A	A	A	GR*D
AIMER	V	I	T	(M
RIRE	V	I	D	RI
RI				RIY
PENSER	V	I	A	P*S
FINIR	V	2	T	FIN
VOIR	V	I	T	VWA
VWA				VWAY
I	F	I	N	DU DICO

DICTIONNAIRE CONSTRUIT :

A				
AIMER	V	I	T	(M
B		I	A	BIL
BEAU	A	A	A	B+N
BON	A			
C		O	M	XEVAL
CHEVAL	N			
D				
E				
EAU	N	R	F	O
ENFANT	N	R	M	*F*
F				
FEMME	N	R	F	FAM
FINIR	V	2	T	FIN
G				
GRAND	A	A	A	GR*D
H				
HOMME	N	R	M	+M
I				
J				
JEUNE	A	R	A	JØN
JOLI	A	R	A	JOLI
K				
L				
M				
N				
O				
P				
PENSER	V	I	A	P*S

	PETIT	A	A	A	PETIT
	PLUME	N	R	F	PLUM
	Q				
	R				
RIY	RIRE	V	I	D	RI
	RI				
	S				
	SOUPE	N	R	F	SWP
	T				
	TRAVAIL	N	O	M	TRAVAY
	U				
	V				
	VIN	N	R	M	V<
VWAY	VOIR	V	I	T	VWA
	VWA				
	W				
	X				
	Y				
	Z				
	l	F	I	N	DU DICO

ENTREE SEMANTIQUE :

FINIR(DEF(PLU(ENFANT)),DEF(BCN(SOUBE)))#

ARBORESCENCE CONSTRUITE :

FINIR (DEF (PLU (ENFANT)) , DEF (BON (SOUPE

APPELS AU DICO :

FIN DEF (->ABSENT DU DICO) B+N PLU SWP (->ABSENT DU DICO) *F*

GENRE, NOMBRE, PERSONNE, PRONOM :

0 1 6 0
1 0 3 0

MORPHOLOGIE ET SYNTAXE :

B+N L)Z. SWP *F* _FINIS\$ LA,

PHRASE='L)Z.*F*_FINIS\$LA,B+NSWP';

SORTIE PHONETIQUE : L)Z*F*_FINIS\$LAB+NSWP

ENTREE SEMANTIQUE :

AI[MER(DEF(CHEVAL),DEF(EAU))]

ARBORESCENCE CONSTRUITE :

AI[MER (DEF (CHEVAL) , DEF (EAU))]

APPELS AU DICO :

(M DICO)	DEF 0	(->ABSENT DU DICO)	XEVAL	DEF	(->ABSENT DU
-------------	----------	--------------------	-------	-----	--------------

GENRE, NOMBRE, PERSONNE, PRONOM :

0	0	3	0
1	0	3	0

MORPHOLOGIE ET SYNTAXE :

LE,	XEVAL	_(M\$	LA,	0
-----	-------	-------	-----	---

PHRASE='LE,XEVAL_(M\$LA,O';

SORTIE PHONETIQUE :

LEXEVAL_(M\$LO

Service de Reprographie

LIMSI

Centre National de la Recherche Scientifique

BP. 30 - 91406 ORSAY - Cedex

Tel: 941 82 50

