

Table des matières

I	La compréhension dans le dialogue Homme-Machine	3
1	Quelques systèmes de DOHM	5
1.1	Introduction	5
1.2	Architecture classique d'un serveur vocal	6
1.3	La reconnaissance de la parole	7
1.3.1	Principes généraux	8
1.3.2	Efficacité de la reconnaissance	9
1.3.3	Mots hors-vocabulaires et erreurs de reconnaissance	10
1.4	Quelques exemples	11
1.4.1	PHILIPS	11
1.4.2	LIMSI : ATIS - MASK - ARISE	12
1.4.3	MIT : TINA	14
1.4.4	TRAINS et TRIPS	15
1.5	Implémenter le caractère pluriel de la communication	16
1.5.1	Interaction et séquentialité	17
1.5.2	Prosodie	17
1.5.3	Multimodalité	18
1.6	Évaluer la pertinence des systèmes	19
1.6.1	Enjeux et problèmes	19
1.6.2	Paradigmes d'évaluation	20
1.7	Conclusion	25
2	La compréhension : enjeux et approches	27
2.1	Des progrès nécessaires	27
2.2	Erreurs de reconnaissance	29
2.3	Méthodes stochastiques	31
2.4	Utilisation de la syntaxe	32
2.4.1	Syntaxe et pouvoir d'expression	32
2.4.2	Syntaxe et complexité	33
2.4.3	Traitements syntaxiques et « fautes »	35
2.5	Conclusion	36

3	Dialogue et communication	39
3.1	Introduction	39
3.2	Pragmatique et actes de langage	40
3.2.1	Les actes de langage	40
3.2.2	Interaction langagière et actes illocutoires collectifs	41
3.3	Modèles de communication verbale	42
3.3.1	Le modèle du code	42
3.3.2	Le modèle inférentiel	43
3.4	Communication et dialogue Homme/Machine finalisé	45
3.4.1	Dialogue Homme/Machine et interaction	46
3.4.2	Compréhension partielle et erreurs de compréhension	47
3.4.3	Définitions de la compréhension en DOHM	48
3.5	Conclusion	50
4	Les spécificités de l'oral spontané	51
4.1	Introduction	51
4.2	Les marques du travail de formulation	52
4.3	Les marqueurs de l'interactivité	54
4.4	Variation et syntaxe de l'oral	55
4.5	Expression orale dans le dialogue Homme/Machine	57
4.6	Conclusion	57
II	Logus	59
5	Principes généraux et première mise en œuvre	61
5.1	Les objectifs	61
5.2	Représentation sémantique des énoncés	62
5.2.1	Introduction	62
5.2.2	Représentation des objets du domaine	64
5.2.3	Reconnaissance et représentation des actes de langage	69
5.2.4	λ -termes typés	72
5.2.5	Graphes conceptuels	73
5.2.6	Pouvoir d'expression	73
5.2.7	Conclusion	75
5.3	Les principes initiaux de l'analyse	76
5.3.1	Grammaires de Montague, λ -termes et analyse partielle	76
5.3.2	Analyses syntaxiques partielles	77
5.4	LOGUS-I : description et évaluations	79
5.4.1	L'architecture de LOGUS-I	79
5.4.2	Évaluation de LOGUS-I et conclusions	81

6	Seconde mise en œuvre	83
6.1	LOGUS-II : principes et formalisme	83
6.1.1	Les principes	83
6.1.2	Le formalisme de représentation des éléments	85
6.1.3	Les étapes de l'analyse	92
6.2	La segmentation en <i>chunks</i>	93
6.2.1	Les rôles de la segmentation	93
6.2.2	Les règles	93
6.2.3	Mise en œuvre	94
6.2.4	Exemple	97
6.3	La connaissance syntaxico-sémantique	97
6.3.1	Dépendances sémantiques	97
6.3.2	Dépendances syntaxico-sémantiques	99
6.3.3	Annexes et conséquences	100
6.4	Les règles syntaxico-sémantiques	101
6.4.1	Définition des règles	101
6.4.2	Les différents niveaux	103
6.4.3	Exemple	107
6.4.4	Noyaux sémantiques	107
6.5	Contexte et dépendances	111
6.5.1	Dépendances entre actes de langage	111
6.5.2	Ébauche d'interprétation contextuelle	113
6.6	Conclusion	113
7	Bilans et perspectives	115
7.1	Évaluations de LOGUS	115
7.1.1	Résultats comparés : précautions et intérêt	116
7.1.2	Résultats globaux	116
7.1.3	La nature des erreurs	118
7.1.4	Les causes des erreurs	119
7.1.5	Erreurs de reconnaissance	121
7.1.6	Conclusion	123
7.2	Stratégies d'analyse : objectifs initiaux et perspectives	123
7.2.1	Robustesse	123
7.2.2	Finesse	126
7.2.3	Généricité	127
7.3	Interrogations existentielles	128
III	Annexes	131
A	Grammaires de Montague et λ-calcul	133
A.1	Grammaires catégorielles	133
A.2	Grammaires de Montague et λ -termes	134
A.2.1	Curry-Howard	134

A.2.2	Calcul de Lambek	134
A.2.3	Les λ -termes simplement typés	135
A.2.4	Calcul de Lambek et sémantique de Montague	136
A.2.5	Exemples	136
A.3	Conclusion	137
B	λProlog	139
B.1	Les formules de λ Prolog	139
B.1.1	Prolog et clauses de Horn	139
B.1.2	Extension de la programmation logique	140
B.1.3	Formules héréditaires de Harrop	140
B.2	Les termes de λ Prolog	142
B.3	La programmation en λ Prolog	143
B.3.1	Le typage	143
B.3.2	Un exemple	143
B.4	Conclusion	144
	Index	153

Table des figures

1.1	Les composants d'un serveur vocal	6
1.2	Exemple de graphes d'entrée et de sortie du système PHILIPS (d'après (Aust et al., 1995))	12
1.3	Les concepts utilisés par l'analyse sémantique de MASK	13
1.4	Requête et schéma sémantique pour l'analyseur MASK	13
1.5	L'arbre d'analyse de l'énoncé: « What street is the Hyatt on? » (d'après (Seneff, 1992b))	14
1.6	Une règle de TINA (d'après (Seneff, 1992b))	14
1.7	L'architecture de TRIPS (d'après (Allen et al., 2001a))	16
1.8	La nouvelle architecture de TRIPS (d'après (Allen et al., 2001b))	18
1.9	Résultats officiels des évaluations ARPA-ATIS en décembre 1994 (Pallet et al., 1995)	22
2.1	Extrait de dialogue du AT&T (Walker et al., 2002)	28
2.2	Extrait de dialogue du LIMSI-ARISE (Lamel et al., 2000)	29
2.3	Exemple d'une représentation sémantique par une structure de traits	34
3.1	Le modèle du code	42
5.1	Exemple de représentation sémantique	72
5.2	Grammaire des termes de la représentation sémantique	74
5.3	Architecture de LOGUS-I	79
6.1	Les principales catégories syntaxiques utilisées dans LOGUS-II	87
6.2	Exemples de catégories syntaxiques	88
6.3	Les principaux rôles sémantiques définis dans LOGUS-II	90
6.4	Quelques exemples de rôles sémantiques	91
6.5	Architectures comparées de LOGUS-I et LOGUS-II	92
6.6	Exemple de segmentation en <i>chunks</i> sur un énoncé complet	98
6.7	Relation d'ordre partiel sur les étiquettes des propriétés	101
6.8	Résultat de l'application des règles de niveau 1	108
6.9	Schéma (simplifié) des règles utilisées au niveau 1	108
6.10	Résultat de l'application des règles de niveau 2	109
6.11	Résultat de l'application des règles de niveau 3	109
6.12	L'ordre sur les rôles sémantiques des noyaux	110
6.13	Représentation sémantique finale de l'énoncé exemple : formule logique.	112

7.1	Résultats comparés de LOGUS-I et de LOGUS-II	118
7.2	Nature des erreurs dans LOGUS-I et LOGUS-II	118
7.3	Causes d'erreurs comparées dans LOGUS-I et LOGUS-II	120
7.4	Les erreurs dues à la complexité de l'organisation syntaxique et sémantique de l'énoncé	121
B.1	<i>Les nouvelles règles de calcul des séquents de λProlog.</i>	141

Introduction

Initialement conçus pour des données numériques, les traitements automatiques se sont très vite étendus aux langues naturelles dans des applications nombreuses et diverses : vérification et correction orthographiques, résumé et indexation de textes, aide à la traduction, etc., donnant lieu à ce qu'il est convenu d'appeler l'*ingénierie des langues*.

Si le langage est le premier des modes de communication humaine, l'expression première du langage est la parole ; l'étude de la communication orale entre l'homme et la machine, le Dialogue Oral Homme Machine (DOHM par la suite), est une branche particulière de l'ingénierie des langues, avec des problèmes qui lui sont spécifiques.

Les systèmes de DOHM actuellement opérationnels sont fortement finalisés : ils ont pour objet des demandes d'information ou la commande d'une tâche. Un utilisateur peut par exemple, en s'exprimant « naturellement », obtenir des horaires de train ou d'avion, des informations météo, etc. Outre le caractère très restrictif du domaine, les dialogues sont en général dirigés par la machine et ne laissent que très peu de place à l'initiative humaine.

L'une des difficultés essentielles à laquelle se heurte le développement de tels systèmes est que l'« *on ne sait pas simuler à l'heure actuelle la compréhension du langage sur ordinateur* » (Danlos, 2002). En effet, les ambiguïtés sont nombreuses dans la langue naturelle. Par ailleurs, et particulièrement en situation de dialogue, l'implicite y est constant : « *le sens linguistique sous-détermine le vouloir dire du locuteur* » (Sperber, 2000). La compréhension des intentions de l'utilisateur du système demande une interprétation en fonction de la situation (but du dialogue) et du contexte du dialogue (prise en compte de l'historique de ce dialogue). À cela s'ajoutent, dans un dialogue oral, les particularités de la langue orale spontanée et les difficultés liées à la reconnaissance du signal acoustique. Dans la plupart des systèmes existants, le caractère très étroit du domaine concerné permet que la compréhension puisse être réduite à la reconnaissance de mots ou de segments clefs qui correspondent à des éléments d'un schéma sémantique prédéterminé.

Le travail présenté dans ce mémoire a pour objectif la conception d'un système de compréhension des énoncés oraux dans un système de DOHM finalisé. Le but est de parvenir à une compréhension des intentions du locuteur qui permette de mettre en œuvre un dialogue plus « naturel » et plus coopératif, pour une tâche moins restrictive que celles habituellement envisagées. Le principe directeur consiste à essayer de tirer le meilleur parti des connaissances linguistiques du matériau à analyser : la parole spontanée en situation de dialogue.

Le système prend en entrée les messages orthographiques délivrés par la reconnaissance vocale. La compréhension est donc conçue comme une traduction d'une liste de mots de la langue naturelle vers une langue cible qu'il convient de définir : assez simple pour permettre une gestion facile du dialogue mais suffisamment précise pour représenter finement les intentions du locuteur. Pour les traitements qui constituent l'analyse proprement dite, les connaissances linguistiques sont utilisées

à double titre :

- Les études sur la langue orale spontanée décrivent les caractéristiques de cette langue ; elles permettent donc d’avoir une meilleure connaissance du matériau que le système doit traiter.
- Comme le sens linguistique n’est « *qu’une indication toujours ambiguë et incomplète du sens voulu par le locuteur* » (Sperber, 2000), la connaissance des modes de communication entre interlocuteurs dans un dialogue est nécessaire pour espérer pallier les insuffisances de ce sens.

Le mémoire est divisé en deux parties :

Première partie : elle est consacrée à l’état de l’art et aux données du problème. Elle comporte une description rapide de quelques uns des systèmes de DOHM existants (chapitre 1), quelques remarques sur les approches utilisées pour la compréhension de la langue orale (chapitre 2), une étude de la communication en situation de dialogue (chapitre 3), une étude des particularités de l’expression orale spontanée (chapitre 4),.

Deuxième partie : elle est consacrée à la description et à l’évaluation du système LOGUS. Son premier chapitre décrit la langue cible utilisée, les objectifs visés, les principes initialement retenus pour les atteindre et une première mise en œuvre de ces principes (chapitre 5). Le chapitre suivant est consacré à la description du système actuel (chapitre 6). Le mémoire se termine par un essai d’évaluation du système et une présentation des nombreux travaux qui permettraient de l’améliorer et de le compléter (chapitre 7).

Première partie

La compréhension dans le dialogue Homme-Machine

Chapitre 1

Quelques systèmes de DOHM

1.1 Introduction

Entretenir une conversation « naturelle » avec une machine dans un dialogue non contraint est actuellement hors de portée et il semble même que les tentatives en ce sens n'aient pas permis d'enregistrer de réels progrès ces trente dernières années (Wilks and Catizone, 2000). Il est de toute façon permis de se demander quel en est exactement l'intérêt. En revanche, la possibilité d'utiliser la langue naturelle pour assurer l'interaction avec une machine que l'on utilise pour accomplir une tâche donnée est évidemment un enjeu intéressant, une « *application à la fois réalisable et utile* » (Gazdar, 1993). Cependant, deux contraintes importantes s'imposent pour qu'un tel « dialogue » puisse être réellement efficace :

- L'utilisation doit pouvoir se faire en temps réel. L'usage de la « langue naturelle » ne se justifie réellement que si les temps d'échange sont comparables à ceux d'une conversation ordinaire.
- L'utilisation ne doit pas nécessiter d'apprentissage. En effet, il est difficile de mettre en avant le caractère « naturel » de l'échange si seuls les « experts » du système sont capables de s'en servir...

Actuellement, les tâches envisagées sont essentiellement des dialogues de commande et des demandes d'information. Dans les deux cas, le dialogue est supposé ne se référer qu'à la tâche en question, dans un micro-monde restreint. Ainsi, depuis une dizaine années, les systèmes de dialogues oraux se multiplient, qu'ils soient prototypes expérimentaux ou applications commerciales.

Cependant, le terme de « dialogue » peut recouvrir des processus très différents. Au minimum, il peut s'agir d'un menu déroulant qui propose à l'utilisateur une série de choix. Les réponses obtenues sont alors très courtes et faciles à analyser et les stratégies de gestion du dialogue extrêmement sommaires. À l'opposé, le dialogue peut offrir un accès naturel à une application, par le biais d'un dialogue se rapprochant d'une conversation humaine et qui laisse à l'utilisateur des possibilités d'initiatives : celui-ci peut lui-même poser des questions et exprimer diverses réactions. Bien évidemment, les énoncés que doit analyser le système sont alors beaucoup plus complexes, ce qui peut conduire à une augmentation du taux d'échecs ; quant à la gestion du dialogue, elle doit s'appuyer sur une meilleure compréhension des intentions de l'utilisateur et prendre en compte les aléas ordinaires des dialogues naturels que sont les « ratés » de la conversation : sa tâche s'en trouve considérablement augmentée.

La suite de ce chapitre présente le fonctionnement général d'un système de DOHM finalisé, la diversité des approches retenues pour la compréhension des énoncés dans quelques-uns des systèmes de DOHM actuellement opérationnels les plus connus et les problèmes qui se posent lorsque l'on cherche à évaluer les performances du module qui opère la compréhension ou la pertinence du système dans son ensemble.

1.2 Architecture classique d'un serveur vocal

Un serveur vocal est composé de plusieurs modules plus ou moins indépendants ; une telle architecture a pour avantage de faciliter la maintenance et l'évolution des systèmes. Elle correspond pour la plupart d'entre eux à la représentation schématique standard présentée dans la figure 1.1.

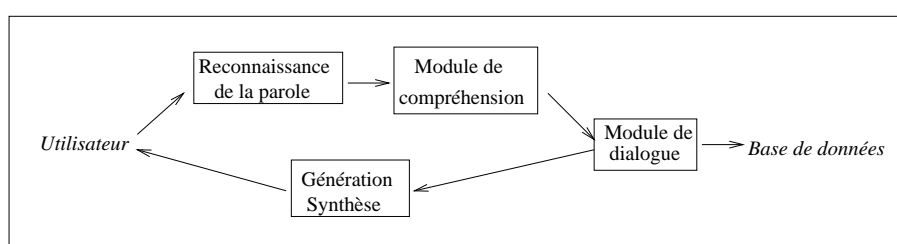


FIG. 1.1 – *Les composants d'un serveur vocal*

Module de reconnaissance de la parole

Le rôle du module de reconnaissance est de transcrire le signal vocal en un message orthographique. En général, sa réponse comporte plusieurs solutions classées suivant leur score et données sous la forme d'un ensemble ordonné de mots.

La tâche du module de reconnaissance (discutée plus en détail au 1.3) est complexe car il doit être capable de s'adapter à des locuteurs très variés, avec des accents régionaux ou étrangers. J. Glass mentionne également le problème posé par les voix d'enfants, pour lesquelles les systèmes n'ont pas forcément été entraînés (Glass, 1999). De plus, beaucoup de serveurs vocaux utilisent le téléphone. Le problème est rendu alors encore plus difficile, de par l'étroitesse des bandes passantes et les perturbations diverses.

Module de compréhension

Le module de « compréhension » prend en entrée le ou les messages orthographiques issus du module de reconnaissance. Il doit transmettre au module de dialogue un message qui puisse rendre compte du sens de l'énoncé en fonction de la tâche normalement poursuivie par le dialogue. Son rôle est donc de traduire un énoncé (retranscrit) de la langue naturelle en une représentation sémantique formelle dont le choix même est essentiel.

Gestionnaire de dialogue

L'étendue du rôle joué par le gestionnaire de dialogue dépend de celui accordé au module de compréhension ; au minimum, si celui-ci lui fournit une interprétation du message qui tient

compte de l'historique du dialogue, le gestionnaire de dialogue doit encore assurer l'interface avec la base de données et proposer les réponses (ou questions) à transmettre à l'utilisateur. Il convient de répondre aux interrogations de ce dernier mais il est également important de lui faire savoir comment le système a interprété sa ou ses requêtes ; ainsi, la possibilité lui est laissée de préciser ou de corriger sa demande. L'élaboration d'un tel module demande donc que soient établis des modèles et des stratégies de dialogue.

Génération de message/synthèse de la parole

La génération « profonde » du message rendu (« que dire ? ») est en général élaborée par le gestionnaire de dialogue ; le générateur proprement dit opère donc essentiellement la génération de surface (« comment le dire ? », avec quels mots etc.) et la synthèse de la parole s'il s'agit d'un système entièrement vocal ; des sorties visuelles sont également possibles.

La compréhension des énoncés dans un tel système est l'objet de ce travail. Il est impossible cependant d'ignorer complètement les modules amont et aval que sont la reconnaissance de la parole et le gestionnaire du dialogue.

En effet, le premier fournit une transcription du matériau sur lequel travaille la compréhension et cette transcription n'est ni complète, ni fidèle. Elle est incomplète car elle filtre les pauses, certaines hésitations et, en général, la prosodie (cf. §1.5.2). Elle est infidèle car elle est sujette à des erreurs : transformation de mots à l'intérieur de son lexique ou transformations de mots qui lui sont inconnus en mots de ce lexique (cf. §1.3.3), élisions, etc. Il est par conséquent nécessaire de connaître la nature des erreurs commises (cf. §1.3).

La compréhension dépend également du module de dialogue pour deux raisons différentes :

- La complexité des énoncés émis par l'utilisateur dépend directement des stratégies de dialogue utilisées. Or, c'est le module de dialogue qui met en œuvre ces stratégies.
- D'autre part, le formalisme choisi pour traduire les énoncés et le niveau de compréhension sont directement dépendants de la tâche que l'on est supposé confier au gestionnaire du dialogue.

Ces interactions entre la compréhension et la reconnaissance d'une part et, d'autre part, entre la compréhension et la gestion du dialogue, soulignent l'arbitraire de ce découpage classique en modules. Certains systèmes intègrent reconnaissance et compréhension. Souvent, le gestionnaire de dialogue est utilisé pour compléter et affiner la compréhension.

1.3 La reconnaissance de la parole

L'existence même des systèmes de DOHM n'a été rendue possible que par les progrès remarquables de la reconnaissance vocale ces vingt dernières années. Cependant, les erreurs que cette reconnaissance génère restent nombreuses et le resteront encore probablement longtemps, sauf nouvelle révolution technologique actuellement imprévisible. Comme les énoncés sur lesquels travaille le module de compréhension en sont issus, il est important de comprendre le fonctionnement de la reconnaissance vocale et la nature des erreurs qu'elle peut engendrer.

1.3.1 Principes généraux

En reconnaissance de la parole¹, la suprématie des approches stochastiques est bien établie et actuellement incontestée. La plupart des systèmes reposent sur une modélisation statistique de la parole. Un modèle linguistique permet d'estimer la probabilité $p(M)$ de toutes les suites finies de mots $M = (m_1, m_2, \dots, m_p)$. Le canal acoustique encode le message M dans le signal X ; il est modélisé par une densité de probabilité $f(X|M)$: probabilité du signal X étant donné le message M .

La reconnaissance de la parole revient à déterminer la suite de mots M qui permet de maximiser la probabilité $p(X)$ d'un signal X observé ; il s'agit donc de déterminer le message M pour lequel la probabilité $p(M) \cdot f(X|M)$ est maximale.

La construction d'un module de reconnaissance correspond à la résolution de trois problèmes :

1. l'élaboration d'un modèle linguistique,
2. l'élaboration d'un modèle acoustique,
3. la réalisation d'un décodeur qui utilise les deux modèles précédents.

Le modèle linguistique

Les modèles linguistiques veulent représenter les régularités de la langue naturelle. La méthode la plus utilisée consiste à approcher la probabilité de la séquence de mots $M = (m_1, m_2, \dots, m_p)$ par le produit des probabilités de chaque mot m_i étant donné son contexte gauche h_i composé des mots qui le précèdent : $p(M) \simeq \prod_{i=1}^p p(m_i | h_i)$. Le contexte gauche h_i est approximé par les $n - 1$ mots qui précèdent le mot m_i (modèle *N-gramme*) et en général, pour des raisons de taille du modèle, la valeur de n est limitée à 1 (modèle bigramme) ou à 2 (modèle trigramme). Ainsi, dans un modèle trigramme, on considère que $p(M) \simeq \prod_{i=1}^p p(m_i | m_{i-2}, m_{i-1})$.

Le modèle acoustique

Les Modèles de Markov Cachés (MMC) sont les plus souvent utilisés pour la modélisation acoustique et leur utilisation est générale pour combiner les modèles linguistiques et acoustiques².

Ces modèles comportent un ensemble S d'états (cachés) et un ensemble O d'observations ; des probabilités décrivent les transitions entre états et les densités d'émission associées à chaque état. La reconnaissance de la parole correspond à la détermination de la suite d'états qui permet de maximiser les émissions observées.

L'élaboration d'un modèle passe par une procédure d'apprentissage. Celle-ci consiste à déterminer l'ensemble de paramètres du modèle de Markov qui permet de représenter au mieux les observations disponibles pour cet apprentissage. Le choix des paramètres eux-mêmes doit être

1. Pour un état de l'art sur ce sujet, on peut par exemple se reporter à (Mariani, 2002).

2. Pour une présentation des modèles de Markov cachés et leur application à la reconnaissance de la parole, on peut par exemple se reporter à (Rabiner, 1989).

adapté aux quantités de données prévues pour cette opération. Par ailleurs, la qualité du modèle obtenu dépend directement de la représentativité de ces données d'entraînement.

Le décodeur

Le décodeur utilise les modèles linguistique et acoustique pour rechercher la séquence de mots qui correspond à la suite d'états la plus probable.

Pour la parole continue à grand vocabulaire, la combinaison des modèles linguistique et acoustique aboutit à des modèles énormes. La reconnaissance demande que soient appliqués des algorithmes capables de traiter de très grands espaces de recherche ; la recherche est en général non exhaustive, limitée à une fraction de l'espace total. Diverses approches sont utilisées pour économiser les temps de calcul et la quantité de mémoire nécessaire.

1.3.2 Efficacité de la reconnaissance

Le paragraphe précédent montre que la qualité de la reconnaissance est limitée par la taille des modèles qui représentent le langage et le canal acoustique ; par exemple, le contexte de l'apparition d'un mot pris en compte est très local puisqu'il ne concerne au mieux, dans les modèles trigrammes, que les deux mots précédents. La qualité des résultats obtenue par le système est liée à ces limitations et plusieurs paramètres interviennent :

La taille du vocabulaire influe directement sur la complexité du modèle à élaborer et la difficulté de l'apprentissage : plus le vocabulaire est étendu, plus les modèles doivent être vastes et les corpus d'apprentissage importants. À cela vient s'ajouter la difficulté de la reconnaissance de la parole continue. Reconnaître quelques mots-clefs isolés est évidemment beaucoup plus simple qu'un flux continu de mots sans limitation du vocabulaire.

La variabilité des locuteurs est également un facteur de difficulté important. Elle se manifeste dans les prononciations, les intonations et les vitesses d'élocution, sans compter les différences entre les voix masculines et féminines et le problème posé par les voix d'enfants. Les données d'apprentissage nécessitent des corpus importants, variés et représentatifs.

La qualité du signal vocal joue également un rôle essentiel. La reconnaissance vocale est évidemment plus difficile dans un environnement « bruité ». Par ailleurs, avec un média tel que le téléphone, la largeur réduite de la bande et la variabilité de la qualité du signal diminuent la fiabilité de l'information véhiculée par ce signal.

Le temps de calcul tolérable intervient également. Une reconnaissance en temps réel signifie que le système doit reconnaître le message pendant la durée du signal, ce qui interdit certaines techniques (passes multiples).

Le dialogue Homme-Machine tel qu'il est envisagé dans ce mémoire est un contexte où la tâche de la reconnaissance est particulièrement difficile : le système doit évidemment être multi-locuteur et accepter la parole continue. Même s'il peut être limité au domaine de l'application, le vocabulaire doit néanmoins être suffisamment vaste pour ne pas entraver le caractère « naturel » de l'expression. Le système doit fonctionner en temps réel. Quant à la qualité du signal vocal, elle est loin d'être assurée ; dans la plupart des systèmes, il s'agit en effet de liaisons téléphoniques.

Les taux d'erreurs sur les mots sont très variables mais s'ils varient couramment entre 5% et 10% pour une dictée vocale, ils peuvent aller jusqu'à 30% lorsque les conditions se détériorent.

En dehors du problème des mots inconnus dont il est question dans le paragraphe suivant, les erreurs portent souvent sur les mots courts ; les mots grammaticaux tels que les prépositions et les déterminants sont donc fréquemment concernés.

1.3.3 Mots hors-vocabulaires et erreurs de reconnaissance

Le modèle de langage utilisé par le module de reconnaissance ne peut contenir tous les mots potentiellement prononcés par les utilisateurs, aussi complet soit-il. Même dans le cas d'une reconnaissance à large vocabulaire, destinée à une dictée vocale, différents phénomènes rendent inévitable l'existence de *mots inconnus*, appelés en général *mots hors-vocabulaires* (Out-of-vocabulary ou OOV) :

- la prononciation inhabituelle d'un mot par ailleurs présent dans son lexique,
- les mots inachevés qui sont relativement fréquents dans la langue orale spontanée,
- les noms propres non répertoriés ou légèrement déformés. La classe des noms propres étant en effet potentiellement infinie, l'exhaustivité y est impossible.
- les expressions régionales ou argotiques,
- etc.

Dans le cas où la reconnaissance est conçue pour un domaine d'application particulier, l'utilisation d'un lexique relativement réduit diminue la taille des modèles nécessaires et peut donc permettre d'améliorer son efficacité. Les mots hors-vocabulaires sont souvent simplement des mots qui sont absents du lexique parce qu'ils sont considérés comme hors-domaine.

En règle générale, lorsque le module de reconnaissance est confronté à un mot ou à un groupe de mots absents de son vocabulaire, il le ou les remplace par le mot ou groupe de mots phonétiquement le plus proche. Ainsi, les mots hors-vocabulaire sont une cause d'erreurs importante de la reconnaissance. Par exemple, I. Bazzi et J. Glass rapportent que le pourcentage de mots hors-vocabulaire auxquels est confronté le système JUPITER (système d'information météo, cf. page 14) est d'environ 2%, et que 13% des énoncés environ en comportent. Or, le taux d'erreurs de reconnaissance sur ces énoncés est considérable, supérieur à 50%, alors qu'il est d'environ 10% pour les autres énoncés (Bazzi and Glass, 2000; Hazen et al., 2002).

Il existe des informations ou des traitements qui permettent d'atténuer les effets de ces erreurs :

- Dans certains modules de reconnaissance, les mots ou groupes de mots reconnus sont accompagnés de scores qui représentent des taux de confiance de la reconnaissance. Ils sont évalués à l'aide de différents paramètres tels que les probabilités d'émission du signal vocal correspondant par rapport aux modèles de langage utilisés, le nombre d'hypothèses engendrées, la différence entre la meilleure de ces hypothèses et les suivantes etc. (Hazen et al., 2002).
- D'autres modules de reconnaissance introduisent des mécanismes destinés à permettre la détection acoustique des mots hors-vocabulaires. Un état de l'art assez complet en est fait dans le mémoire de thèse de C. Bousquet (Bousquet-Vernhettes, 2002). Par exemple, I. Bazzi et J. Glass cherchent à inclure un modèle acoustique des mots hors-vocabulaires dans le modèle de langage utilisé par la reconnaissance (Bazzi and Glass, 2001). Le problème est de détecter les mots hors-vocabulaires sans pour autant désigner comme tels les mots qui ne le

sont pas (fausses-alertes) ; pour tenter de le résoudre, les auteurs attribuent un coût aux mots hors-vocabulaires. Appliqué à JUPITER (cf. §1.4.3), le meilleur de leurs quatre modèles permet une détection de 70% des mots hors-vocabulaires pour un taux de fausses-alertes de seulement 3,2%.

Message vocal-message orthographique

En dehors du problème des erreurs occasionnées par la reconnaissance, il convient également de remarquer que la traduction d'un message vocal en un message orthographique n'est pas sans conséquence. Les perturbations ainsi créées peuvent paraître anodines par rapport aux erreurs de la reconnaissance mais elles méritent néanmoins d'être mentionnées. En effet, les indices orthographiques ne sont pas équivalents à ceux supportés par la phonétique :

- En général, l'orthographe véhicule des indices supplémentaires, phonétiquement indétectables. Ainsi, à l'oral, les marques de pluriel sont généralement portées seulement par le déterminant alors qu'à l'écrit, elles apparaissent systématiquement dans les terminaisons des noms communs et dans les conjugaisons ; un autre exemple est celui de la préposition « à » que les énoncés retranscrits distinguent du verbe avoir conjugué « a ».
- Si le module de reconnaissance n'a pas commis d'erreurs sur les indices ajoutés, ils ne gênent en rien la compréhension. Par contre, certains indices phonétiques qui pourraient lui être utiles disparaissent dans les retranscriptions orthographiques : par exemple, la prononciation du *s* final permet de distinguer si « *y a plus de chambres dans cet hôtel* » signifie « *il n'y a plus de chambres dans cet hôtel* » ou « *il y a davantage de chambres dans cet hôtel* ».

Ainsi, les erreurs de reconnaissance, l'absence de la prosodie et la non-coïncidence des indices orthographiques et phonétiques font que le message orthographique retransmis au module de compréhension est une image brouillée du message vocal envoyé par le locuteur.

1.4 Quelques exemples

Ce paragraphe montre comment est envisagée l'analyse des énoncés dans quelques uns des serveurs vocaux les plus connus actuellement opérationnels.

1.4.1 PHILIPS

Le système PHILIPS (Aust et al., 1995) est l'un des premiers à avoir donné lieu à des applications grand public. Son architecture générale est en tout point conforme à la description précédente. Elle répond au désir affiché par ses concepteurs de pouvoir adapter le système à différentes applications. La stratégie adaptée pour la compréhension consiste à privilégier avant tout la robustesse, dans la perspective de très nombreuses erreurs de reconnaissance (jusqu'à 30% d'erreurs de mots). L'analyseur sémantique prend en entrée un graphe de mots étiqueté par des probabilités. L'analyse ne prend en considération que les mots ou séquences de mots qui correspondent à des « concepts » de l'application, modélisés sous la forme d'une paire attribut-valeur. Par exemple, l'expression « *to Hamburg* » est représentée par le concept <destination,Hamburg>. L'analyse se fait en plusieurs étapes :

- Le graphe de mots est transformé en un graphe de concepts étiquetés par des probabilités.

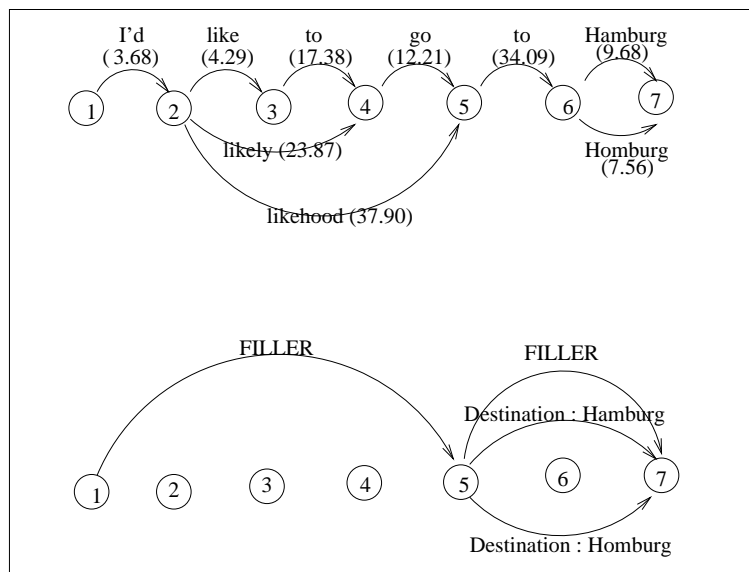


FIG. 1.2 – Exemple de graphes d'entrée et de sortie du système PHILIPS (d'après (Aust et al., 1995))

- Le graphe de concepts est complété pour relier entre eux les concepts reconnus et créer des concepts dits vides entre les concepts non adjacents.
- La requête est déterminée à partir de la séquence de concepts la plus probable.

La représentation sémantique de l'énoncé est donc un graphe de concepts. La figure 1.2 représente un graphe de mots pris en entrée et le graphe de concepts rendu par l'analyseur.

L'analyse est basée sur les règles d'une grammaire sémantique, hors-contexte et probabiliste. Au départ, les règles de la grammaire sont générées manuellement sur un grand volume de données. Leurs probabilités sont ensuite estimées sur un corpus de phrases étiquetées automatiquement par la grammaire elle-même.

Sur ce système, un certain nombre d'applications ont été implémentées telles que, par exemple :

- horaires de trains en Allemagne et pour la langue allemande,
- information sur les horaires de la Lufthansa,
- guide en langue anglaise pour la restauration dans la région de Boston.

1.4.2 LIMSI : ATIS - MASK - ARISE

Les tâches couvertes par les systèmes LIMSI-ATIS, MASK et ARISE sont assez proches :

- Dans LIMSI-ATIS (Air Travel Information Service), un système d'interaction vocale permet à l'utilisateur d'obtenir des informations issues d'un guide des compagnies aériennes américaines et canadiennes (Minker, 1996).
- MASK (Multimodal Multimedia Automated Services Kiosk) traite des demandes d'information sur les voyages ; le système francophone de MASK, développé par le LIMSI-CNRS permet d'avoir des informations sur les trains : horaires, services à bord et prix (Lamel et al., 1995).

<i>Concept</i>	<i>Requête exemple</i>
train-heure	<i>quels sont les <u>horaires</u> des trains allant de Paris à Lyon</i>
tarif	<i>quel est le <u>prix</u> du billet</i>
correspondance	<i>quels sont les <u>changements</u></i>
type	<i>quel est le type du train qui arrive à 20 heures</i>
réserver	<i>je veux <u>réserver</u> une place dans le train de 8 heures 30</i>
service	<i>quelles sont les <u>prestations</u> offertes dans ces trains</i>
réduction	<i>qu'est-ce qu'un <u>billet Joker</u></i>

FIG. 1.3 – Les concepts utilisés par l'analyse sémantique de MASK

<u>Requête</u> <i>je veux aller demain matin de Paris à Marseille en passant <u>par Lyon</u></i>	
<u>Schéma sémantique</u> < train-heure >	
ville-départ	: Paris
ville-arrivée	: Marseille
ville-escale	: Lyon
jour-relatif	: demain
période-journée	: matin

FIG. 1.4 – Requête et schéma sémantique pour l'analyseur MASK

- ARISE (Automatic Railway Information Systems for Europe) est un service de renseignements par téléphone sur les horaires de train (Lamel et al., 2000).

Les analyseurs sémantiques de ces systèmes sont fondées sur le formalisme des *frames grammar*, inspirées des *grammaires de cas* (Fillmore, 1968; Bruce, 1975). Les requêtes sont représentées par des schémas. Le schéma qui correspond à un énoncé est identifié par certains mots. Les attributs du schéma (les cas) correspondent à des segments contextuels qui peuvent éventuellement être identifiés par des marques syntaxiques locales (prépositions, déterminants, etc). La figure 1.3 donne les concepts utilisés par l'analyseur sémantique de MASK. Les mots soulignés sont ceux qui permettent l'identification du concept. La figure 1.4 donne un exemple de schéma sémantique pour ce même système ; la relation conceptuelle *train-heure* est munie de cinq attributs : *ville-départ*, *ville-arrivée*, etc. Seuls les mots en caractères gras sont pris en compte pour l'analyse de l'énoncé. Les mots soulignés sont les identificateurs des attributs.

Des analyseurs stochastiques ont été réalisés pour ces systèmes et ont fait l'objet d'évaluations qui ont permis de les comparer aux analyses par règles. Pour ces analyses stochastiques, le jeu de règles déterminé manuellement est remplacé par des connaissances acquises automatiquement sur les correspondances entre des mots et des étiquettes sémantiques ; ces connaissances sont les paramètres d'un modèle markovien (cf. §1.3.1). Les résultats obtenus ont été comparables, voire très légèrement supérieurs à ceux obtenus par les analyses par règles. Les avantages et les inconvénients des deux méthodes sont discutés au 2.3.

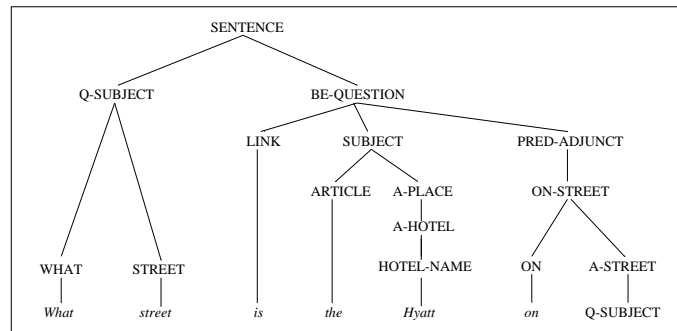


FIG. 1.5 – L'arbre d'analyse de l'énoncé : « *What street is the Hyatt on?* » (d'après (Seneff, 1992b))

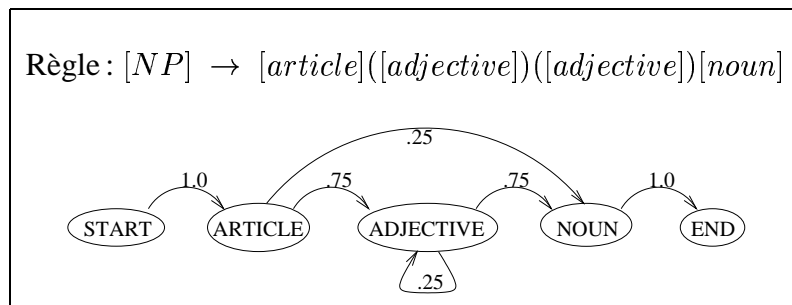


FIG. 1.6 – Une règle de TINA (d'après (Seneff, 1992b))

1.4.3 MIT : TINA

Le système TINA (Seneff, 1992b) est un système de compréhension d'énoncés de l'oral spontané, développé par le MIT et financé par la DARPA³. Le principe fondamental est que l'analyse, pour être efficace, doit associer syntaxe et sémantique (Seneff, 1998). Le résultat rendu est un arbre, supposé par hypothèse unique ; les nœuds de l'arbre correspondent à des catégories qui peuvent être syntaxiques ou sémantiques (un exemple d'arbre d'analyse est donné à la figure 1.5).

L'analyse elle-même est descendante, basée sur des règles syntaxiques auxquelles viennent s'ajouter des contraintes sémantiques. Les règles, qui sont celles d'une grammaire hors-contexte, sont fabriquées directement à partir de l'analyse d'énoncés type. Pour aider au choix des différentes règles lors de l'analyse, chacune d'entre elles a la forme d'un graphe probabiliste comme dans l'exemple de la figure 1.6. Par ailleurs, des catégories sémantiques hiérarchisées sont implémentées dans TINA : par exemple *RESTAURANT* est considérée comme une sous-catégorie des catégories *BUILDING* et *PLACE*. Elles permettent d'introduire des filtres sémantiques. Ainsi, l'énoncé « *What street is the Hyatt on?* » (« *Sur quelle rue se trouve le Hyatt* ») est considéré comme un énoncé correct (dont l'arbre d'analyse est représenté à la figure 1.5) alors que le filtre sémantique interdit à l'énoncé « *What street is Cambridge on?* » (« *Sur quelle rue se trouve Cambridge* ») d'en être un.

Cependant, à l'usage, la contrainte d'une analyse syntaxique complète des énoncés s'est révélée trop forte. Les énoncés rejetés sont donc soumis à une deuxième analyse, partielle, qui ne prend en

3. Defense Advanced Research Projects Agency

compte que les segments conceptuels « porteurs de sens » (Seneff, 1992a).

D'autre part, S. Seneff souligne la difficulté de concevoir des règles qui associent syntaxe et sémantique tout en assurant la gestion d'une connaissance sémantique du domaine concerné (Seneff, 1998).

Ces grammaires ont été implémentées pour différentes applications : guide touristique pour Boston et sa région (système VOYAGER, (Glass et al., 1995)), informations et réservations aéronautiques (système PEGASUS, (Zue et al., 1994)), informations météorologiques (système JUPITER, (Zue et al., 2000)), informations sur des annonces pour l'achat de voitures d'occasions (système WHEELS, (Meng et al., 1996)), guide pour la restauration (Seneff and Polifroni, 1996).

GALAXY-II permet aux utilisateurs un choix plus large de requêtes grâce à une reconnaissance du sous-domaine concerné. La requête est alors réorientée vers tel ou tel serveur déterminé pour y être traitée (Seneff et al., 1999).

1.4.4 TRAINS et TRIPS

Les concepteurs de TRAINS, puis de TRIPS (The Rochester Interactive Planning System) se proposent de mettre en œuvre des stratégies de dialogue qui puissent laisser aux utilisateurs une grande part d'initiative, et qui puissent s'appliquer à des tâches variées dans des domaines divers. Ils fondent leurs espérances sur les deux principes suivants (Allen et al., 2000) :

- Si la compétence dialogique à mettre en œuvre dans un dialogue ayant pour but une tâche déterminée est complexe, elle est néanmoins beaucoup plus simple que celle requise dans un dialogue humain en général (cf. chapitre 3).
- La plus grande partie de la complexité à maîtriser tant pour la compréhension des énoncés que pour les stratégies de dialogue est indépendante de la tâche.

Dans TRAINS (Allen et al., 1996), la tâche consiste à déterminer un itinéraire ferroviaire. L'utilisateur a devant les yeux une carte sur un écran où figurent les villes et les lignes. Il peut dialoguer avec le système pour déterminer le meilleur itinéraire, en évitant les encombrements et les interruptions de trafic.

L'analyse des énoncés est ascendante et basée sur des règles de grammaire qui font appel simultanément aux catégories syntaxiques et sémantiques des constituants. Le résultat de l'analyse est une suite d'actes de langage. Par exemple, l'énoncé « *Okay now let's take the last train and go from Albany to Milwaukee* » (« *D'accord maintenant le dernier train pour aller d'Albany dans le Milwaukee* ») fait l'objet d'erreurs de reconnaissance et transmis au module de compréhension comme étant la suite de mots : « *Okay now I take the last train in go from Albany to is* » (« *D'accord maintenant je prends le dernier train vais d'Albany vers est* »). Le module de compréhension rend une suite de trois actes :

1. une *CONFIRMATION* (« *Okay* »),
2. un acte *NEUTRE* (indéterminé) dont le contenu est « prendre le dernier train » (« *now I take the last train* »),
3. une *REQUÊTE* de trajet en provenance d'Albany (« *in go from Albany to is* »).

Alors que la compréhension est incomplète et même partiellement erronée, elle permet néanmoins la poursuite du dialogue.

Les résultats obtenus par TRAINS ont été très encourageants (90% de dialogues réussis).

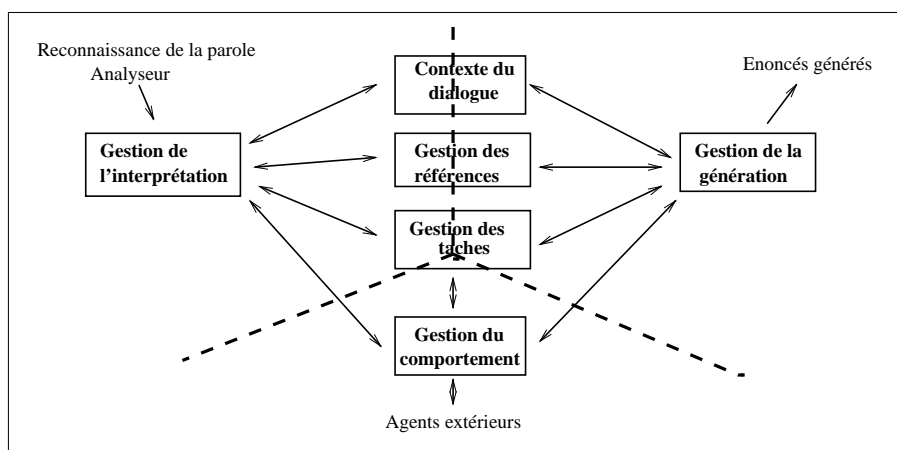


FIG. 1.7 – L'architecture de TRIPS (d'après (Allen et al., 2001a))

Avec TRIPS, les concepteurs de TRAINS se proposent de construire un système de dialogue pragmatique très interactif et adaptable à diverses applications : TRIPS-PACIFICA concerne l'évacuation d'une île en cas de cyclone, TRIPS-CPOF le déploiement de troupes militaires, TRIPS-KITCHEN le plan d'une cuisine, etc. (Allen et al., 2000). L'une des méthodes utilisées pour accroître à la fois la portabilité et la qualité de l'interaction homme-machine est l'adoption d'une architecture plus complexe et très modulaire représentée dans la figure 1.7. Les différents composants sont répartis dans trois aires de fonctionnalité qui correspondent respectivement à l'interprétation, le comportement et la génération. Chaque aire comporte un module principal qui coordonne le comportement des autres et transmet les messages.

L'analyse des énoncés est très proche de celle faite dans TRAINS. La grammaire est composée d'un noyau indépendant du domaine de l'application. Le problème est celui de l'adaptation de cette grammaire à des domaines particuliers. Des prédicats spécifiques à ces domaines sont définis. Par exemple, la grammaire générique contient une classe verbale *MOVE*. Or, dans l'application PACIFICA, les actions de transport jouent un rôle essentiel : *TRANSPORT* est un prédicat spécifique. Aussi, dans cette application, la classe des verbes *MOVE* exige l'existence d'un *instrument* de type véhicule et d'un *theme* qui soit un objet mobile (transportable). La réduction du nombre des énoncés qui peuvent être ainsi analysés avec ces contraintes supplémentaires augmente en fait l'efficacité de l'analyseur et permet de réduire les ambiguïtés.

1.5 Implémenter le caractère pluriel de la communication

Comme on le verra dans la chapitre 3, les linguistes et les philosophes qui s'intéressent à la communication verbale insistent sur le caractère pluriel de celle-ci. Dans les conversations, les partenaires ne communiquent pas seulement par les propos qu'ils échangent mais également par une certaine gestuelle, telle que l'orientation du corps ou les regards. Même au niveau du langage proprement dit, la communication ne se limite pas à des phrases échangées. Elle se manifeste éga-

lement au travers de ce que les linguistes appellent « phatiques » ou « régulateurs » (« *ah bon* », « *oui* », « *tu vois* » etc.) par lesquels les différents partenaires se manifestent l'intérêt et l'attention qu'ils se portent l'un à l'autre (Kerbrat-Orecchioni, 1990). En particulier, le fonctionnement des tours de parole n'est pas un processus séquentiel d'actes monologiques mais une « *synchronisation interactionnelle* » : les locuteurs respectent le système des tours, qui obéit au principe de « *minimization of gap and overgap* »⁴. Alors même qu'on lui parle, le présumé récepteur manifeste sa participation à la conversation par la gestuelle et les régulateurs.

Certains systèmes de dialogue homme-machine essaient d'intégrer ces particularités du dialogue humain.

1.5.1 Interaction et séquentialité

Actuellement, la plupart des systèmes de dialogue Homme-Machine sont conçus selon le principe d'un strict respect de la séquentialité des traitements. L'utilisateur parle ; pendant que son message est analysé, interprété et que la réponse est générée, le système n'accepte aucune perturbation extérieure. Les dialogues ainsi obtenus sont en ce sens assez éloignés des dialogues humains.

J.Allen, G. Ferguson et A. Stent (Allen et al., 2001a) citent plusieurs extraits du corpus Monroe (Stent, 2000) qui montrent des situations concrètes d'interactivité des locuteurs dans un dialogue « pragmatique » :

- régulateur émis par le récepteur entre les différents éléments d'une énumération (« *mm-hm* », « *okay* »),
- réponse interrompue par un complément ajouté à la requête,
- réponse qui correspond à un simple accusé de réception de la requête (« *yeah in one minute* »).

Les auteurs proposent une nouvelle architecture pour le système TRIPS (cf. figure 1.8) qui permet aux trois principaux composants : l'interpréteur, le générateur et le gestionnaire du comportement de travailler d'une manière asynchrone. Par exemple, le générateur peut envoyer une réponse alors même que le gestionnaire du comportement n'a pas encore décidé ce qu'il fallait faire. Cette architecture permet ainsi un dialogue plus interactif et plus « naturel » pour les utilisateurs.

1.5.2 Prosodie

Certaines analyses associent la syntaxe et la sémantique mais très peu jusqu'à présent utilisent la prosodie. Pourtant, celle-ci est depuis longtemps considérée comme une ressource importante pour la compréhension des énoncés oraux. Lorsque P. Blache énonce que « *l'interprétation d'un énoncé se fait grâce à des éléments d'information pouvant provenir, de façon redondante ou non, de chacun des composants* », il inclut la prosodie dans l'ensemble de ces composants qui contient également la morphologie, la syntaxe, la sémantique et la pragmatique. Selon lui, l'absence de la prosodie comme élément d'analyse tient à l'organisation même de l'information linguistique, qui tend à une représentation séparée et hiérarchisée des différents composants (Blache and Christo, 2002). D'ores et déjà cependant, certains travaux se proposent de montrer que la prise en compte de la prosodie peut améliorer la reconnaissance automatique de la parole et même la compréhension,

4. Les silences et les chevauchements sont réduits au minimum dans une conversation suivie.

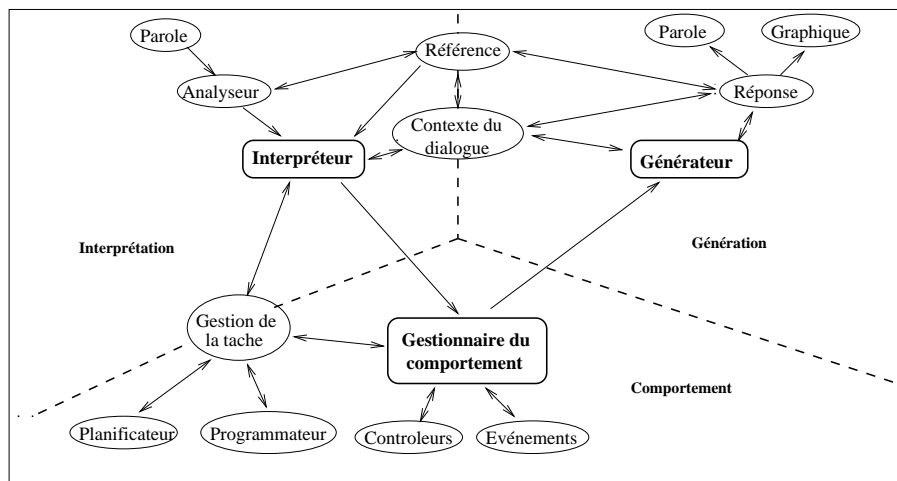


FIG. 1.8 – La nouvelle architecture de TRIPS (d'après (Allen et al., 2001b))

du moins lorsqu'elle est fondée sur des règles stochastiques (Shriberg et al., 1998; Shriberg and Stolcke, 2002).

1.5.3 Multimodalité

La multimodalité des systèmes est un domaine en plein essor et sans aucun doute prometteur. L'un des arguments avancés est que combiner différents modes permet de compenser les faiblesses de chacun d'entre eux (Oviatt, 2000). Ces dernières années ont vu se développer les études théoriques sur la multimodalité et un certain nombre de systèmes sont déjà opérationnels. On peut citer par exemple, et sans aucune prétention d'exhaustivité, les systèmes suivants :

- GEORAL (Siroux et al., 1997) est un système d'information touristique. Il permet de combiner les approches visuelles (l'utilisateur peut observer une carte sur son écran), vocales (l'utilisateur peut parler en langue naturelle) et tactiles (l'utilisateur peut désigner un point ou une zone sur la carte qu'il a sur son écran).
- GALAXY-II (Seneff et al., 1999) joue le rôle de super-structure pour intégrer les différents systèmes d'information de la DARPA tels que JUPITER (informations météorologiques), PEGASUS (information sur les horaires des avions entre les principales villes des États-Unis), MERCURY (réservation sur les vols d'avions) et VOYAGER (trafic routier et aide à la navigation routière). Les utilisateurs peuvent utiliser un mode vocal (informations téléphoniques) ou une interface graphique (navigation sur le Web) ou combiner les deux modes d'utilisation.
- MUST est un projet EURESCOM qui s'est déroulé en 2001 et 2002 qui avait pour objet l'étude des possibilités de mise en œuvre de services multimodaux sur les petits terminaux mobiles (Boualem and Col., 2002). Un tel projet démontre l'enjeu commercial représenté désormais par la multimodalité.
- REA est un robot androïde qui peut communiquer par le geste, le regard et la parole en jouant le rôle d'un agent immobilier (Bickmore and Cassell, 2003). La gestion du dialogue ne laisse

aucune initiative à son interlocuteur.

Les exemples qui viennent d'être présentés montrent la variabilité des approches adoptées pour la compréhension des énoncés. Comparer ces diverses solutions suppose une évaluation des systèmes ; celle-ci fait l'objet du paragraphe suivant.

1.6 Évaluer la pertinence des systèmes

1.6.1 Enjeux et problèmes

Procéder à l'évaluation des systèmes de DOHM est une tâche difficile mais néanmoins indispensable, qui constitue une partie importante des efforts consacrés à leur conception et à leur réalisation. Les résultats attendus des évaluations sont de deux natures différentes, parfois difficiles à concilier :

- On attend d'une évaluation qu'elle permette de montrer l'efficacité du travail et de valider les choix théoriques effectués. En ce sens, elle ne peut être véritablement opérationnelle que si elle autorise des comparaisons avec d'autres systèmes, basés sur des principes différents.
- On peut également attendre d'une évaluation qu'elle mette en évidence les points forts et les points faibles de l'approche choisie ou de sa réalisation. Ce pouvoir de diagnostic est particulièrement intéressant lors de la phase de conception du système.

D'après les définitions du dictionnaire, « *évaluer* » un système c'est « *fixer la valeur* » de ce système ; cette valeur peut être considérée comme une mesure de « performance globale ». Elle ne peut être interprétée que par des comparaisons directes ou indirectes, par rapport à des références connues ou implicites. Pour une évaluation qui vise essentiellement à établir un classement par ordre d'efficacité de différents systèmes - en supposant, bien sûr, qu'ils soient comparables -, il peut sembler que le résultat idéal soit une mesure unique ; en effet, un classement peut alors être établi par une comparaison directe des mesures résultats. Cependant, il semble évident qu'une telle mesure ne peut à elle toute seule permettre de détecter les faiblesses d'un système. Pour autoriser un diagnostic, une évaluation requiert au minimum que les résultats rendus soient des séries de mesures et que de plus, chacune d'entre elles correspondent à des situations spécifiques. La comparaison des différentes séries de résultats permet alors d'évaluer les points faibles et les points forts des différentes approches des systèmes évalués. L'étude de ces séries de résultats permet également d'appréhender le comportement général d'un système donné et de surveiller les évolutions de ses performances au fur et à mesure de sa mise au point.

D'une manière générale, on désigne sous le nom de « boîte noire » (*black box*) les modes d'évaluation qui cherchent à mesurer la pertinence des systèmes de DOHM dans leur globalité alors que ceux prennent en compte les modules internes du système sont dits « boîte transparente » (*glass-box*). Dans la mesure où la compréhension est en général un module relativement autonome d'un système de DOHM, on peut penser que son évaluation (boîte transparente) est plus simple que celle du système dans sa globalité (boîte noire). Cependant, ses performances ne sont indépendantes ni du domaine et de la tâche, ni des formes de dialogue envisagées : concevoir son évaluation pose différents problèmes.

- La qualité de la compréhension peut être estimée en comparant la représentation sémantique obtenue avec la représentation « attendue ». Sa mesure correspond alors à la définition d'une

métrique sur l'espace des représentations sémantiques possibles.

Par exemple, lorsque la représentation sémantique repose sur la notion de schémas sémantiques où les différentes requêtes sont représentées par des séquences de paires attributs-valeurs, l'évaluation de la compréhension peut se définir par comptage des erreurs sur chaque ligne. On peut obtenir une typologie des erreurs en les classant suivant trois catégories : les insertions, les substitutions et les élisions. Ainsi, dans un système qui donne des renseignements sur les horaires de train, les types d'erreurs peuvent être l'élision d'une ville ou d'une gare, l'insertion d'un élément marquant un horaire, une substitution entre les valeurs d'un triplet marquant une date, etc. Les progrès du système sont alors évalués en étudiant le taux des erreurs considérées comme les plus coûteuses (celles dont les conséquences sont les plus graves pour la réussite du dialogue), à savoir les insertions et substitutions qui concernent les villes d'arrivée ou de départ et les dates (Rosset, 2000).

- Le point précédent fait apparaître que la définition d'une mesure de la qualité de la compréhension est liée à l'espace des représentations sémantiques concerné. Par ailleurs, il a déjà été souligné que le choix de cette représentation sémantique est un élément essentiel pour la bonne conduite du dialogue et qu'il dépend de la stratégie de dialogue envisagée. Comme chaque système utilise un formalisme de représentation sémantique qui lui est propre, la comparaison des performances entre différents systèmes pose un problème difficile : comparer des performances évaluées avec des métriques différentes ne va pas de soi.

Lorsque les tâches et les domaines des systèmes coïncident, une solution consiste à imposer une représentation commune. Le risque existe alors d'avantager les systèmes évalués dont le formalisme initial est proche du formalisme demandé.

- Enfin, la finesse et la pertinence de la compréhension ne peuvent être réellement estimées que par rapport à l'utilisation que peut en faire le module de dialogue. L'efficacité de la compréhension ne peut donc être totalement prise en compte qu'intégrée à l'ensemble du système de dialogue ; le problème se pose donc d'un mode d'évaluation « boîte noire » ou « boîte transparente ».

1.6.2 Paradigmes d'évaluation

Évaluations subjectives et objectives

Plusieurs paradigmes ont été proposés pour évaluer les systèmes de DOHM ou leurs composants⁵.

- Les tâches auxquelles sont destinés les systèmes envisagés dans ce mémoire consistent essentiellement à fournir des renseignements à leurs utilisateurs ; ceux-ci peuvent donc être considérés comme étant les meilleurs juges de la pertinence du système. Ainsi, un indice de satisfaction fourni par un échantillon représentatif d'utilisateurs peut sembler a priori un bon critère d'évaluation. Une telle évaluation dite *subjective* se révèle dans la pratique très difficile à opérer ; S. Rosset décrit différents problèmes rencontrés lors d'une expérience de ce type menée au LIMSI (Rosset, 2000) : variabilité intra-sujet (influence de l'adaptation au

5. L. Devillers, H. Maynard et P. Paroubek présentent un état de l'art sur ce sujet dans (Devillers et al., 2002).

système), variabilité inter-sujets, représentativité des sujets choisis par rapport aux utilisateurs potentiels du système (sans compter la difficulté de définir un profil de ces utilisateurs potentiels) « sérieux » des sujets recrutés (soumis à la tentation de « jouer » avec le système), etc. Ces difficultés de mise en œuvre expliquent que l'on ait généralement recours à des évaluations d'un autre type.

- Les évaluations dites *objectives* sont centrées sur les résultats des systèmes en dehors des impressions que peuvent en avoir leurs utilisateurs.

Par exemple, si l'on cherche à évaluer la qualité du dialogue, le problème se pose de déterminer les différents paramètres à prendre en compte. Classiquement, les évaluations tentent de prendre en compte des critères tels que la pertinence des informations obtenues par rapport à la requête initiale, le nombre d'échanges nécessaires pour y parvenir, la durée du dialogue, etc. Il va de soi que les valeurs prises par ces paramètres sont largement dépendantes du domaine concerné et des tâches attendues. Par ailleurs, chacun d'entre eux pris séparément ne constitue qu'un indice du bon déroulement de la négociation. La question se pose alors de savoir comment les intégrer pour obtenir une évaluation globale.

Comme il est admis que la finalité des systèmes de DOHM est le service rendu à leurs utilisateurs, des solutions ont été imaginées pour contourner la difficulté de la mise en œuvre des évaluations subjectives. Ainsi, le système PARADISE (PARAdigm for Dialogue System Evaluation) du MIT correspond à la recherche d'une évaluation subjective à partir d'évaluations objectives (Walker et al., 1997). L'objectif final est en effet de parvenir à mesurer la satisfaction des utilisateurs. Différentes mesures de performances sont effectuées sur le système : accomplissement de la tâche, coût du dialogue, etc., qui correspondent à une évaluation objective de celui-ci. Des tests effectués sur des échantillons d'utilisateurs permettent de connaître la corrélation entre ces performances et la satisfaction des usagers. Par régressions linéaires, le système fabrique alors une fonction qui permet de combiner les différents paramètres objectifs obtenus pour obtenir un nombre qui est censé représenter le plus probablement l'indice de satisfaction des utilisateurs. Ce paradigme a été utilisé pour comparer des systèmes de dialogue portant sur différentes tâches et pour surveiller les évolutions de leurs performances (Walker et al., 2002; Bonneau-Maynard et al., 2000).

Malgré tout, aussi intéressant que soit ce type d'approche, les évaluations objectives demeurent les plus simples à effectuer et les plus courantes, en particulier pour les systèmes de compréhension. Les paragraphes suivants en décrivent quelques paradigmes.

Les évaluations ATIS

Entre 1990 et 1994, un vaste programme d'évaluation est mené par la DARPA américaine. Le groupe MADCOW (Multi-site ATIS Data Collection Working group) propose un paradigme d'évaluation qui a pour objet de comparer les systèmes du groupe ATIS ; leur tâche consiste à répondre aux demandes d'information sur les transports aériens. Pour contourner la difficulté liée à l'absence de représentation sémantique commune, l'évaluation porte sur les « bonnes réponses » rendues par le système à la requête proposée. Une réponse est classée comme « bonne » lorsqu'elle s'inscrit entre deux réponses de référence construites manuellement : une réponse minimale qui contient les informations explicitement demandées et une réponse maximale qui intègre des informations supplémentaires qui peuvent être raisonnablement incluses dans la réponse. Les réponses erronées sont affectées d'un poids double de celui affecté aux absences de réponses.

Site	AT&T	CMU	MIT-LCS	SRI	BBN	UNISYS	MITRE
% d'erreurs	3,8	3,8	4,5	7,0	9,4	23,6	30,6

FIG. 1.9 – Résultats officiels des évaluations ARPA-ATIS en décembre 1994 (Pallet et al., 1995)

L'un des avantages essentiels est l'objectivité du procédé : tous les systèmes sont soumis à une même batterie de tests (pour la plupart obtenus par la technique du Magicien d'Oz), ce qui permet des comparaisons directes. Cependant et bien qu'il ait été l'un des plus utilisés (Minker, 1998a), ce système d'évaluation comporte des inconvénients soulignés assez rapidement par les porteurs du projet eux-mêmes (Hirschman et al., 1992; Hirschman, 1998) :

- La réponse du système peut être correcte malgré une compréhension erronée. Interpréter la correction de la compréhension par celle de la réponse introduit un biais méthodologique faible mais néanmoins non nul.
- La rapidité de la réponse n'est pas prise en compte.
- Cette évaluation ne permet de comparer entre eux que des systèmes qui portent sur une même tâche dans un même domaine.
- Enfin et surtout, si elle donne une mesure des performances globales des systèmes, elle ne permet pas d'analyser les causes des erreurs comptabilisées ; elle n'a pas de réel pouvoir de diagnostic.

Les évaluations ATIS ont joué un rôle important dans le développement de la Compréhension Automatique de la Parole (CAP) : elles ont été les premières campagnes importantes du domaine et elles ont mis en évidence les avancées de celui-ci.

Le tableau de la figure 1.9 donnent les résultats officiels des évaluations en décembre 1994 : les erreurs sont données en pourcentages. Elles ne portent que sur les requêtes qui peuvent se comprendre en dehors du contexte de dialogue (les autres énoncés, compréhensibles en contexte ou jugés hors domaine ayant été systématiquement rejetés des batteries de tests).

Il est difficile de tirer des conclusions sur les qualités et les défauts des différentes approches au vu de ces résultats. Parmi les systèmes obtenant les meilleures performances globales, figurent le système du AT&T où la compréhension repose sur un modèle de Markov caché dont les états correspondent à des unités sémantiques élémentaires et TINA (cf. page 14) fondé sur des règles de grammaire probabilistes.

Ainsi donc, le manque de généralité et de prédictivité constituent les deux points faibles des évaluations de type ATIS. Le paradigme DCR a pour but de les corriger.

Le paradigme DCR (Demande-Contrôle-Réponse)

Inspiré du paradigme DQR (Déclaration, Question, Réponse) utilisé pour l'évaluation de la compréhension de textes, le paradigme DCR a été proposé dans le cadre de l'action de recherche « dialogue oral » de l'AUF (Antoine and Caelen, 1999; Antoine et al., 2002a).

Ce paradigme permet de contourner le problème des représentations sémantiques communes. L'objectif est une évaluation de type diagnostic, centrée sur les erreurs dues aux spécificités de la parole spontanée : le système est évalué par une série de tests, classés selon des phénomènes linguistiques déterminés.

Chaque test est composé de trois éléments :

1. Le premier est l'énoncé sur lequel porte le test proprement dit ;
2. Le second $\langle C \rangle$ est un énoncé dit de contrôle qui doit en principe être compris sans risque d'erreur par le système et qui est donc aussi simple que possible ; il reprend, d'une manière positive ou négative l'information présente dans $\langle D \rangle$.
3. Le troisième $\langle R \rangle$ est la référence, un booléen qui correspond au résultat de la comparaison entre D et C.

Par exemple,

$\langle D \rangle$ « *Quels sont les vols en partance de Lyon pardon d'Athènes* »

$\langle C \rangle$ « *Quels sont les vols en partance de Lyon* »

$\langle R \rangle$ [FAUX]

Le système construit séparément les représentations sémantiques qui correspondent à $\langle D \rangle$ et $\langle C \rangle$. Leur compatibilité sémantique est évaluée par un booléen. L'évaluation du test est positive si le résultat de l'étude de la compatibilité correspond à la valeur de la référence.

Ainsi ce processus d'évaluation est basé sur des mesures objectives. Il est relativement générique par rapport à la tâche et au système : chaque série de tests peut en effet être destinée à évaluer les performances du système face à un phénomène particulier. Le nombre de tests nécessaires pour pouvoir déterminer les causes d'erreurs essentielles en font une méthodologie assez lourde, mais à fort pouvoir de diagnostic ; elle constitue donc un complément intéressant aux évaluations de type ATIS, mais reste difficile à mettre en œuvre.

L'évaluation « par défi »

Cette méthodologie d'évaluation des systèmes de compréhension a été proposée par le groupe de travail GT du GDR-I3 du CNRS (Antoine et al., 2002a). Sa conception répond à quatre objectifs essentiels :

1. L'évaluation est basée sur des critères objectifs.
2. Elle doit permettre de faire un diagnostic précis concernant les qualités et les défauts des systèmes, quels que soient le domaine concerné et la tâche envisagée.
3. Elle se veut relativement légère par rapport aux évaluations de type DCR.
4. Une analyse commune des erreurs détectées doit permettre de mutualiser les expériences des différents participants.

Cinq systèmes ont été testés lors de la première campagne d'évaluation organisée par le GTI3⁶. Les domaines ou les tâches excluant d'emblée des séries de tests communs, le protocole adopté a été le suivant : pour chacun des cinq systèmes, vingt énoncés initiaux représentatifs de la tâche et du domaine ont été proposés. Ensuite, à partir des énoncés initiaux d'un système donné, les quatre autres participants ont élaboré chacun quinze énoncés dérivés de chaque énoncé initial. Chacun des systèmes a donc disposé d'un corpus spécifique de 1200 tests, avec des types de phénomènes très variés. Chaque développeur a ensuite effectué les tests sur son propre système. Enfin, les différents résultats ont été analysés, comparés et discutés lors d'une séance de travail commune.

6. un système du CLIPS (réservation hôtelière), un système de l'IRIT (horaires de train), un système du LIMSI (renseignements ferroviaires), et deux systèmes du VALORIA (ROMUS et LOGUS : renseignement touristique).

Cette première campagne a fait apparaître quelques problèmes liés à cette méthodologie d'évaluation :

- Les énoncés dérivés sont des oraux simulés. Le problème de la connaissance que l'on peut avoir des modes d'expression des utilisateurs potentiels sera évoqué ultérieurement (cf. §4.5) ; au mieux, la technique du magicien d'Oz⁷ donne un instantané du mode d'expression dans une situation donnée. Mais, elle est assez lourde à mettre en place et l'évaluation « par défi » se veut légère.
- Le choix des énoncés initiaux est essentiel ; il n'est pas facile de s'assurer qu'ils permettent une couverture suffisante du domaine à tester.
- Il est parfois difficile de décider si un énoncé dérivé doit être considéré comme hors-tâche ou hors-domaine et s'il doit être rejeté ; et l'on peut défendre l'idée, pour les énoncés douteux, qu'il est intéressant de tester les systèmes sur la frontière de leur domaine de compétence.
- Des comparaisons chiffrées entre systèmes sont rendues difficiles par l'hétérogénéité des difficultés et des phénomènes testés.

Malgré ces réserves, cette campagne a fourni aux participants une occasion de confronter leurs points de vue qui étaient d'autant plus différents que les systèmes étaient consacrés à des tâches variées. Elle a en même temps atteint son objectif essentiel, qui était de permettre à chacun des participants de juger des points forts et des points faibles de son système et de ses approches.

Le paradigme PEACE

Le paradigme d'évaluation PEACE (Paradigme d'Évaluation Automatique de la CompréhEn-sion) proposé par le LIMSI a pour objectif de proposer une évaluation objective et de type diagnostique de la compréhension, hors-contexte et en contexte (Devillers et al., 2002).

La possibilité de porter un diagnostic sur les systèmes testés est atteinte grâce à des batteries de tests, classées suivant des phénomènes linguistiques déterminés.

L'évaluation en contexte (de dialogue), jugée indispensable par des observations selon lesquelles elle serait un élément déterminant de la satisfaction des utilisateurs, nécessite une prise en compte de l'historique du dialogue : une paraphrase obtenue par correction manuelle des énoncés précédant l'énoncé test modélise le contexte du dialogue.

Ainsi, PEACE permet à la fois une évaluation objective, en contexte et de type diagnostique. Cependant, les mesures des performances proposées sont basées sur le principe de représentations sémantiques suivant les schémas sémantiques prédéfinis avec des paires <attributs-valeurs> classiquement utilisés dans les analyses sélectives ; elles risquent donc d'être essentiellement bien adaptées à ces systèmes.

La prochaine campagne d'évaluation MEDIA lancée dans le cadre du projet TECHNOLANGUE du ministère doit utiliser ce paradigme.

En conclusion, les évaluations propres aux systèmes de compréhension sont indispensables pour permettre de juger les approches et leur réalisation ; elles sont d'autant plus nécessaires que l'analyse des énoncés oraux reste un problème largement ouvert, pour lesquels des stratégies très différentes peuvent être envisagées. On peut sérieusement mettre en doute leur capacité à classer

7. La technique du magicien d'Oz consiste à faire parler un utilisateur à un interlocuteur humain tout en lui faisant croire qu'il s'adresse à une machine.

les systèmes de compréhension par ordre de pertinence. Leur principal intérêt est donc lié aux possibilités qu'elles offrent de faire des systèmes testés un diagnostic pertinent et précis.

1.7 Conclusion

Le rapide tour d'horizon précédent et les quelques exemples de dialogues oraux qui ont été présentés montrent que l'on sait construire des systèmes de dialogues oraux qui répondent aux deux conditions nécessaires requises initialement : donner des réponses en temps réel et pouvoir être utilisés par des novices. Les performances générales de ces systèmes sont acceptables quant aux taux de réussite des dialogues et de satisfaction de leurs utilisateurs. Ces indéniables succès témoignent des progrès significatifs du DOHM.

Cependant, même si ces avancées sont très encourageantes, elles ne doivent pas cacher que l'élaboration de systèmes plus complexes, qui puissent permettre un dialogue plus « naturel » et moins contraint ou qui soient restreint à une tâche moins circonscrite demande des progrès très importants dans différents domaines.

Concernant la compréhension, plusieurs remarques s'imposent lorsqu'on examine les approches adoptées dans les systèmes présentés :

- Tous les systèmes de compréhension s'appuient sur la sémantique : ils utilisent des connaissances liées au domaine de l'application. Cependant, cette utilisation prend des formes diverses : formes sémantiques attendues (LIMSI), règles d'analyse incluant la sémantique (TINA), etc. Le problème est souvent posé de savoir comment concilier l'utilisation de telles connaissances avec la généralité des systèmes (Seneff, 1998; Dzikovska et al., 2003).
- Les modules de compréhension rendent en résultat une représentation sémantique de l'énoncé sous une forme très variable. Elle est en effet liée à la répartition des tâches entre le module de compréhension et le module de dialogue, à la définition de la tâche couverte par le système ainsi qu'aux stratégies de dialogue qui président à son développement : graphe de concepts (PHILIPS), schéma sémantique (LIMSI), arbre d'analyse mêlant syntaxe et sémantique (TINA), suite d'actes de langage (TRAIN'S).
- Les différences se manifestent également dans les stratégies d'analyse adoptées : analyse ascendante partielle (TRAIN'S), analyse descendante complète (TINA), sélection de segments clefs (LIMSI).

D'une manière un peu surprenante, la compréhension en tant que telle ne semble pas avoir été au cœur des recherches dans le domaine du DOHM ces dix dernières années : le système TINA est utilisé dans les systèmes du MIT et il semble donner satisfaction ; les concepteurs de TRIPS disent que l'analyse des énoncés s'appuie sur les mêmes principes que ceux utilisés dans TRAINS'95. Deux causes peuvent être avancées pour expliquer ce fait :

1. Jusqu'à présent, la compréhension n'est pas apparue comme le maillon faible des systèmes. Le taux des erreurs de reconnaissance excluait de toute manière une analyse un peu fine d'une part et d'autre part, la faible complexité des systèmes ne faisait pas apparaître la nécessité d'un raffinement de l'analyse. Une meilleure gestion du dialogue a pu apparaître comme plus prioritaire.
2. Le paragraphe sur l'évaluation montre que comparer les performances liées aux différentes approches est difficile ; dans les cas où les évaluations ont cherché à faire des comparai-

sons directes entre les systèmes testés, elles n'ont pas permis de démontrer clairement la supériorité de l'une ou l'autre de ces approches.

Ainsi, les méthodes à utiliser pour la compréhension des énoncés émis par les utilisateurs dans un dialogue Homme-Machine finalisé ne sont pas déterminées. Le chapitre suivant étudie quelques points particuliers liés à cette diversité des approches en relation avec les progrès nécessaires de la compréhension automatique de la parole pour le développement du DOHM.

Chapitre 2

La compréhension : enjeux et approches

2.1 Des progrès nécessaires

La plupart des systèmes actuellement opérationnels couvrent des tâches très précises dans des domaines étroits. De plus, contrairement à ce que l'on pourrait attendre dans un dialogue dit « naturel », ils sont essentiellement dirigés par la machine et ils «*obligent l'utilisateur à penser tout à la fois à ce qu'il veut faire et à la façon de l'exprimer* » (Pierrel and Romary, 2000).

Pour que les systèmes de DOHM puissent devenir des outils efficaces et conviviaux, des progrès sont nécessaires et deux paramètres essentiels sont à prendre en considération :

1. l'étendue du domaine de l'application et la complexité de la tâche¹ ,
2. l'initiative laissée à l'utilisateur dans le dialogue.

Domaine, tâche et ambiguïté :

Pour une tâche bien circonscrite dans un domaine étroit, les requêtes possibles peuvent être modélisées par des structures sémantiques relativement simples. Ainsi, deux ou trois segments clefs peuvent suffire à la compréhension d'une requête d'informations météo : lieu, date, etc. Il en est de même des horaires de trains ou d'avions : date, destination, lieu de départ.

Il semble que la complexité du domaine et de la tâche influe assez peu sur les spécificités de l'expression orale spontanée (Sabah, 1997). Par contre, si la tâche devient plus complexe, les représentations sémantiques correspondantes le deviennent également et le sens des énoncés est donc plus difficile à déterminer. Par ailleurs, une extension du domaine de l'application peut induire une augmentation de l'ambiguïté : le simple repérage de quelques segments conceptuels ne suffit plus à la compréhension. En effet, la plupart d'entre eux deviennent sujets à des interprétations différentes ; ils ne prennent leur sens qu'en fonction du contexte de l'énoncé lui-même ou du dialogue. Par exemple, dans une application où la tâche est celle des horaires de train, « *de Toulouse* » est reconnu comme un lieu de provenance. Par contre, le même syntagme replacé dans un domaine plus large tel que le renseignement

1. Le *domaine* désigne le micro-monde concerné. La *tâche* correspond aux opérations que peut accomplir le système. Par exemple, dans le domaine ferroviaire, la tâche peut consister à donner des informations sur les horaires de trains ; elle peut être élargie à des demandes de réservations ou à tout autre renseignement complémentaire.

touristique ne peut faire l'objet d'une telle « compréhension », étant donnée la multiplicité des interprétations possibles dont il peut faire l'objet.

Les autres composants du système dépendent également de la taille du domaine ; les modules de reconnaissance de la parole sont d'autant plus performants qu'ils utilisent un lexique restreint. Les élargissements du domaine d'application conduisent à un problème très complexe : une reconnaissance de la parole continue multilocuteur avec, de surcroît, un vocabulaire étendu (cf. page 7). On peut remarquer également qu'une extension du domaine induit une multiplication ou une plus grande complexité des schémas sémantiques potentiels que le gestionnaire de dialogue doit savoir gérer.

Initiative laissée à l'utilisateur :

Les systèmes décrits dans le chapitre précédent se veulent en général à initiatives partagées. Selon les cas, cette dénomination recouvre des degrés d'initiative bien différents qui sont souvent peu explicités. Les figures 2.1 et 2.2 sont deux extraits de dialogues de systèmes à initiatives partagées ; le premier illustre les capacités du gestionnaire de dialogue AMELIA du système AT&T de la DARPA (Walker et al., 2002) ; le second est un extrait du système du LIMSI-ARISE (Lamel et al., 2000).

S1	Welcome... What airport would you like to fly out? (<i>Bienvenue... De quel aéroport voulez-vous partir ?</i>)
U2	I need to go to Dallas (<i>je souhaite aller à Dallas</i>)
S3	Flying to Dallas. What departure airport was that? (<i>un vol pour Dallas. Quel est votre aéroport de départ ?</i>)
U4	From Newark on September the 1st. (<i>Newark le premier septembre.</i>)
S5	What time would you like to travel on September the 1st to Dallas from Newark? (<i>À quelle heure souhaitez-vous faire le voyage de Dallas à Newark le premier septembre ?</i>)

FIG. 2.1 – Extrait de dialogue du AT&T (Walker et al., 2002)

Le premier système est en effet capable d'accepter les initiatives de l'utilisateur puisqu'il interprète correctement la réponse U2 qui n'est pas une réponse attendue à la question S1. Cependant, et pour autant que l'on puisse en juger sur deux extraits aussi courts, le second système semble d'emblée proposer davantage d'initiatives à son utilisateur que le premier, comme l'indiquent l'absence de question en S4 et la question très ouverte de S6.

Si une plus grande part d'initiatives laissée à l'utilisateur rend évidemment plus complexe la gestion du dialogue, elle influe également directement sur le niveau de compréhension exigé. Dans un dialogue totalement contraint, les énoncés de l'utilisateur sont censés correspondre à des intentions attendues du système : le « comprendre » consiste donc uniquement à reconnaître le sens de cet énoncé dans une liste de sens prédéfinis. Il en va tout autrement lorsque l'utilisateur a le droit de prendre des initiatives : la compréhension de ses intentions et plus généralement le sens de son message doit se déduire des énoncés eux-mêmes, replacés dans

S1 :	Quels renseignements désirez-vous ?
U2 :	Je voudrais l'horaire du premier train de Paris à Grenoble vendredi prochain
S3 :	Veillez patienter ...
S4 :	Vendredi 26 septembre, le premier train part à 6 heures 48 et arrive à 9 heures 47
U5 :	et un peu plus tard ?
S6 :	Vous avez un train partant de Paris à 7 heures qui arrive à Grenoble à 10 heures 59 avec un changement à Lyon-Part-Dieu le 26 septembre. Qu'en pensez-vous ?
U7 :	Est-ce qu'il y a un bar dans ce train ?
S8 :	Oui, vous avez un bar dans ce train
U9 :	Bon, je réserve
...	...

FIG. 2.2 – *Extrait de dialogue du LIMSI-ARISE (Lamel et al., 2000)*

leur contexte. Une analyse beaucoup plus fine est donc nécessaire.

Ces remarques montrent que dès que l'on cherche à élargir l'éventail des tâches et le domaine d'application ou dès que l'on tente d'améliorer la qualité du dialogue en offrant une plus large initiative à l'utilisateur, les problèmes se multiplient et ils concernent alors les différents composants du système : reconnaissance de la parole, compréhension des énoncés et gestion du dialogue.

Pierrel et Romary insistent sur la nécessité des progrès de la CAP pour dépasser les limites actuelles du DOHM : « *il est nécessaire d'aller au-delà d'une simple compréhension d'énoncés pour parvenir à une interprétation contextuelle en fonction, entre autres, de l'historique du dialogue et de l'application, et des intentions et présuppositions de l'utilisateur* » (Pierrel and Romary, 2000).

La suite du chapitre étudie trois problèmes liés aux différentes approches utilisées pour rendre compte du sens des énoncés en fonction de ces exigences : le traitement des erreurs de reconnaissance, l'utilisation de traitements statistiques et la prise en compte de critères syntaxiques pour l'analyse des énoncés.

2.2 Erreurs de reconnaissance

Il va de soi que les mots mal reconnus peuvent poser de gros problèmes à la compréhension et qu'ils peuvent aller jusqu'à rendre le message orthographique incompréhensible, même pour un expert humain. Ils constituent indéniablement l'une des causes majeures des erreurs des systèmes de dialogue oraux Homme-Machine. Ces erreurs sont le plus souvent traitées « en creux » par les systèmes : elles sont compensées par la robustesse de l'analyse et la prise en compte de critères sémantiques.

Le problème des mots absents du lexique de la reconnaissance de la parole et des erreurs de reconnaissance qu'ils provoquent a déjà été évoqué dans le chapitre précédent (cf. page 10). Peu

d'études existent sur le problème des mots-inconnus au niveau de la compréhension². Les traitements proposés dépendent des informations que l'on peut attendre du système de reconnaissance utilisé (cf. page 10).

- En dehors de toute information venant du module de reconnaissance, C. Bousquet propose une détection des mots inconnus au niveau du système de compréhension (Bousquet-Vernhettes, 2002) (le système, stochastique, a pour domaine d'application est les horaires de train). Le but recherché est que la représentation sémantique d'un énoncé tel que « *je voudrais aller à jour* » contienne l'information [VILLE_ARRIVEE = HORS_VOCA]. L'approche est fondée sur l'étude du contexte, en particulier des prépositions. Un examen du corpus montre que la catégorie « Villes » correspond à 54,4% des mots inconnus et que seulement 7,4% des mots inconnus ne sont ni des villes, ni des mots inutiles. Son indépendance du système de reconnaissance utilisé rend cette approche très intéressante.
- Les travaux de T.J. Hazen, S. Seneff et J. Polifroni sur le système JUPITER (informations météo) (cf. page 15) sont un exemple d'utilisation des taux de confiance rendus par le système de reconnaissance SUMMIT. Les règles de grammaire du système TINA sont modifiées pour tenir compte de ces scores : certains mots à faible taux de confiance sont analysés en tant que *villes inconnues* à condition que le contexte de leur apparition indique clairement qu'il s'agit effectivement de noms de villes. Différents tests réalisés par les auteurs montrent que la meilleure solution est de rejeter les mots reconnus avec un taux de confiance trop bas ; elle permet en effet dans certains cas d'analyser le reste de l'énoncé plutôt que de le rejeter dans son ensemble.
- Les travaux de G. Chung utilisent la détection des mots inconnus par le module de reconnaissance. Ils concernent également le système JUPITER. Ils intègrent à la compréhension la notion de *ville inconnue* détectée préalablement par le système de reconnaissance (Chung, 2000). Le principe consiste à ajouter des règles de grammaire au système de compréhension TINA, par apprentissage : dans les énoncés tests utilisés, une ville sur dix est simplement remplacée par le marqueur *inconnu*. Il va de soi que ce traitement conduit à une réduction notable des taux d'erreurs.

Dans tous les systèmes cités, les noms de ville constituent une catégorie de mots inconnus particulièrement fréquente et importante par les erreurs générées : donner un horaire de train ou une information météo sans ce type d'indication n'a pas de sens. On peut se demander si les traitements utilisés ne sont pas rendus possibles par le caractère très finalisé de ces applications. Par exemple, dans un domaine plus large tel que celui du renseignement touristique, les noms propres peuvent désigner des catégories d'objets divers tels que hôtels, restaurant, rue, cinémas, etc. La détection des mots inconnus, et surtout la possibilité de leur attribuer une étiquette sémantique semble alors nettement plus difficile.

2. Pour un état de l'art sur cette question, on pourra se reporter au mémoire de thèse de C. Bousquet (Bousquet-Vernhettes, 2002).

2.3 Méthodes stochastiques

J. van Benthem fait remarquer que les énoncés que nous émettons sont composés pour une grande part d'éléments figés et souvent répétés et que le traitement automatique des langues a longtemps ignoré cette propriété, qui laisse penser que l'analyse de la langue peut s'appuyer sur des considérations statistiques (van Benthem, 2000).

Comme elles reposent sur des modèles dont les paramètres sont déterminés par apprentissage, il convient, pour que les méthodes stochastiques soient efficaces, de pouvoir disposer de corpus importants et représentatifs.

Après leur mise en service, certains systèmes de DOHM tels que ceux du MIT peuvent recevoir plusieurs millions d'appels chaque mois, ce qui permet de disposer de très larges corpus de données (Hastie et al., 2002). La masse des données ainsi disponibles et leur suprématie en reconnaissance de la parole suscitent une expérimentation des méthodes stochastiques dans différents domaines : compréhension des énoncés, gestion du dialogue (Young, 1999; Young, 2002) et même stratégie du dialogue (Walker et al., 2002) ou analyse de ses échecs (Hastie et al., 2002).

Dans le domaine de la compréhension, les méthodes stochastiques sont largement utilisées mais, s'il est incontestable qu'elles peuvent donner de bons résultats, le cadre applicatif où elles peuvent être réellement efficaces est indéterminé. Jusqu'à présent, elles ont été testées avec succès dans des domaines très étroits pour des tâches très restrictives telles que la recherche d'horaires de train (Bousquet-Vernhettes, 2002) ou d'avion³ (Minker, 1999). Les avantages qui sont mis en avant sont les suivants (Minker, 1999; Young, 1999; Young, 2002) :

- Maintenance plus simple du système : l'effort humain est limité à un étiquetage de données, tâche en principe moins lourde que la mise au point et la maintenance de règles de grammaire.
- Meilleure portabilité : les méthodes d'apprentissage dépendent peu de l'application et de la langue. Elles sont donc aisément portables d'une application à l'autre ou d'une langue dans l'autre. Cette propriété et la précédente permettent l'implantation de systèmes assez robustes, pour des coûts relativement bas.
- Règles plus « réalistes » : les experts humains ont tendance à surestimer l'importance de certains phénomènes linguistiques relativement rares. Les méthodes stochastiques sont par essence faites pour accorder à chaque phénomène rencontré sa « vraie place », statistiquement parlant.

Au nombre des inconvénients constatés, on peut citer les suivants :

- La qualité de l'implémentation dépend directement de la quantité des corpus disponibles et de la représentativité de ces corpus.
- Les règles stochastiques semblent avoir quelques difficultés à rendre correctement compte de certains phénomènes dus au modes de production de l'oral spontané, tels que les répétitions, les auto-corrrections et les reprises (Minker, 1998b). Il semble en effet que ces phénomènes ne répondent que très peu à des critères statistiques précis.
- Par nature, les règles stochastiques ne sont pas aptes à permettre le traitement d'énoncés que l'on peut qualifier de « marginaux » parce qu'ils sont à la limite du domaine ou de la tâche

3. Dans les évaluations ATIS, (cf. page 21), les résultats des systèmes utilisant des méthodes stochastiques sont comparables, voire légèrement supérieurs mais les différences ne sont pas significatives.

ou parce qu'ils correspondent à des demandes de faible fréquence.

Il semble que fonder la compréhension des énoncés uniquement sur des règles stochastiques soit une méthode efficace pour des systèmes qui correspondent à l'exécution de tâches simples et répétitives, pour lesquelles il est assez facile de disposer de données suffisamment représentatives. Par définition même, les systèmes stochastiques privilégient la compréhension des énoncés qui correspondent à des requêtes courantes, émis sous les formes les plus fréquentes. On peut alors se demander si une utilisation exclusive de critères statistiques permet d'instaurer un dialogue où une plus grande part d'initiative est laissée à l'utilisateur, condition nécessaire pour que le dialogue devienne un peu plus « naturel ».

Par exemple, les systèmes de dialogue tels que ceux qui sont étudiés par l'équipe de Rochester (TRIPS, cf. §1.4.4) ont pour objet des tâches sensiblement plus complexes. Le dialogue doit tenir compte d'un monde contextuel en évolution permanente et la stratégie de dialogue laisse une grande initiative à l'utilisateur. Leurs concepteurs prétendent que les méthodes stochastiques ne peuvent pas rendre compte efficacement de la complexité du langage naturel pour ce type de compréhension (Allen et al., 2001a). Minker lui-même, qui par ailleurs met fortement en avant l'intérêt des méthodes stochastiques, affirme que les critères statistiques ne devraient pas être seuls pris en compte pour la compréhension des énoncés.

Des réserves émises sur la compréhension en « tout stochastique » avec des modèles *N-grammes* (cf. §1.3.1) n'excluent pas pour autant la possibilité d'intégrer des éléments statistiques dans les analyses syntaxiques ou sémantiques, mais selon des méthodes qui ont encore été peu étudiées. L'utilisation ou non de critères statistiques n'est probablement pas un problème en soi. Les méthodes statistiques sont très diversifiées et elles s'associent d'ores et déjà à des formes d'analyse qui peuvent être très différentes : analyse syntaxico-sémantique dans TINA ou analyses sélectives dans LIMSI-ARISE.

Cette coopération des méthodes sémantiques et statistiques est certainement un axe de recherche encore trop peu exploré qui peut conduire à des résultats intéressants.

2.4 Utilisation de la syntaxe

Un autre point essentiel sur lequel diffèrent les systèmes de compréhension est celui de la stratégie utilisée pour la prise en compte de la syntaxe. Ce paragraphe pose le problème du pouvoir d'expression ajouté par la syntaxe et des méthodes utilisables pour en rendre compte.

2.4.1 Syntaxe et pouvoir d'expression

Compréhension sans syntaxe : les *pidgins* sont les premiers modes d'expression verbaux qui émergent lorsque des individus sont amenés à communiquer entre eux sans utiliser leur langue maternelle. Dans ces langages très sommaires, un énoncé correspond à une juxtaposition de mots où la grammaire est apparemment absente ; les mots sont groupés en constituants sémantiques, dans un certain ordre. J.-L. Dessalles, qui s'interroge sur l'existence d'une telle compétence protolinguistique chez les ancêtres de l'*homo sapiens*, essaye d'analyser la *protosémantique* que peut véhiculer un tel langage, qualifié de *protolangage* (Dessalles, 2000).

Selon cet auteur, les mots véhiculent des images et, juxtaposés, ces mots suscitent la combinaison d'images mentales qui correspondent à la reconstitution de scènes. Ainsi par exemple, « *chat paillasson* » correspond à l'image d'un chat sur un paillasson. L'ordre des éléments constitue la seule indication possible pour contrôler la combinaison des images ; c'est ainsi que « *fille voler argent partir* » suscite la représentation d'une scène dynamique (un mini-film) où une fille vole de l'argent puis s'en va. Il est évident que ces associations de scènes contrôlées par la seule juxtaposition des différents constituants est un processus peu sûr. Apparemment cependant, un tel langage permet de représenter des perceptions évoquées par des objets concrets.

Dans un système de questions/réponses à une base de données tel que ceux envisagés, la plus grande partie des énoncés correspond à des requêtes - ou à des actes de langage s'y rapportant - portant sur des objets concrets. Il est donc légitime de se demander si le sens des énoncés émis par les utilisateurs appartient au pouvoir d'expression d'un *pidgin*. Si cette hypothèse était vérifiée, une connaissance sémantique des objets du domaine à laquelle vient s'ajouter la prise en compte des seuls mots lexicaux et de l'ordre dans lequel ils sont énoncés permettrait d'appréhender la façon dont doivent se composer les syntagmes. Une analyse basée sur ces principes devrait donc suffire, du moins dans la plupart des cas, à reconstituer le sens du message émis par l'utilisateur du système. Or, même si cette intuition se vérifie pour un bon nombre d'énoncés présents dans les corpus, elle s'avère erronée dans d'autres cas. L'une des causes apparaît évidente : les locuteurs sont habitués à parler une autre langue qu'un simple *pidgin* ; ils utilisent naturellement les mécanismes offerts par la syntaxe pour préciser les associations entre les syntagmes. Les énoncés peuvent contenir des représentations d'objets complexes ou plusieurs propositions, coordonnées ou subordonnées. Réduire un tel énoncé de l'oral spontané à une simple juxtaposition de ses mots lexicaux ne permet pas sa compréhension : il devient impossible de reconstituer certains objets complexes obtenus par coordination et également très difficile d'appréhender comment s'articulent les propositions coordonnées ou subordonnées.

Les marquages syntaxiques

La syntaxe utilise deux procédés complémentaires pour préciser comment les différents syntagmes doivent être composés afin de reconstruire le sens de l'énoncé : les marquages morphologiques et les systèmes de branchement entre syntagmes. Ces deux procédés sont utilisés conjointement dans la plupart des langues mais leurs importances relatives varient. D'une manière générale, les marquages morphologiques tels que les déclinaisons autorisent une plus grande variabilité dans l'ordre des syntagmes. En français, ces marquages, sans être totalement absents, sont relativement réduits et l'ordre des syntagmes joue un rôle très important (langue dite à *ordre fixe*) ; cependant, ces caractéristiques sont encore plus affirmées dans la langue anglaise.

2.4.2 Syntaxe et complexité

On a vu que deux éléments conditionnent la difficulté de la compréhension : l'étendue du domaine et de la tâche et la part d'initiatives laissée à l'utilisateur du système. Ces deux paramètres se conjugent pour accroître l'espace des significations attendues et sa complexité, et donc pour augmenter les ambiguïtés.

- Lorsque l'utilisateur n'a droit à aucune initiative, que ses énoncés ne sont que des réponses

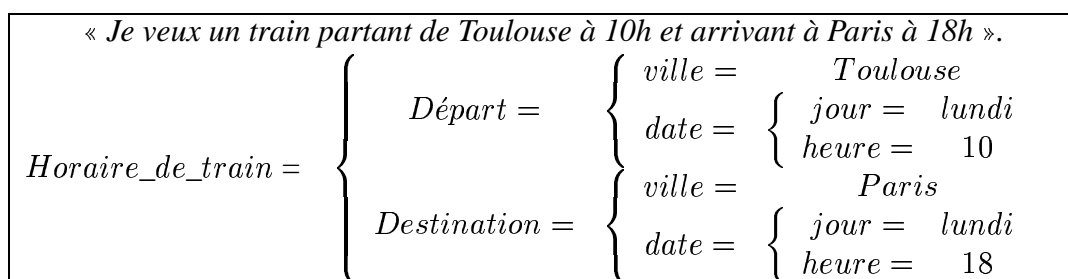


FIG. 2.3 – Exemple d'une représentation sémantique par une structure de traits

à des questions posées, et lorsque, de plus, les concepts liés à l'application ne sont pas trop complexes, une simple analyse par mots-clés est suffisante : la syntaxe peut être totalement ignorée.

- Dans la plupart des systèmes actuellement opérationnels décrits dans le chapitre 1, la tâche et les stratégies de dialogue sont sensiblement plus complexes. Cependant, la tâche reste assez précise et simple pour que la représentation sémantique des énoncés puisse être contenue dans des cadres sémantiques ou schémas pré-déterminés (*frames* en anglais). Il s'agit souvent de structures de traits comme dans la figure 2.3 empruntée à C. Bousquet (Bousquet-Vernhettes, 2002).

Pour ce type de systèmes plusieurs méthodes d'analyse sont en concurrence où la syntaxe est plus ou moins présente.

Certaines analyses dites parfois « sélectives » comme celle du système MASK (cf. page 12) consistent à sélectionner dans l'énoncé les segments conceptuels « signifiants », à identifier le schéma sémantique concerné et à remplir les attributs correspondants. Le dialogue qui s'en suit peut correspondre à des confirmations ou corrections des attributs ou avoir pour but de compléter le schéma. Ces analyses utilisent des indices syntaxiques locaux tels que prépositions et déterminants pour une bonne identification des segments conceptuels. Leur principale qualité est la robustesse.

À l'inverse, un système de compréhension tel que TINA (cf. page 14), utilisé par les systèmes de dialogue du MIT, tente d'abord une analyse complète des énoncés, avant d'avoir recours à une analyse moins contraignante et plus robuste, qui est sélective.

On ne sait pas au juste jusqu'à quel point une telle analyse syntaxique complète est réellement utile dans ces types de systèmes. Les bonnes performances de TINA ne permettent pas de conclure quant à l'efficacité de son analyse syntaxique car l'on ignore quelle est la part des énoncés qui font l'objet de la deuxième technique d'analyse utilisée. J. Glass par exemple (Glass, 1999) prétend qu'elle n'est certainement pas nécessaire dans un système tel que JUPITER, dont la tâche est assez simple.

- L'exemple de TRIPS a déjà été évoqué à propos des méthodes stochastiques. La mouvance du contexte et la multiplication des schémas sémantiques potentiels interdisent la pré-détermination de schémas destinés à modéliser la représentation sémantique des énoncés. Les concepteurs insistent particulièrement sur l'importance qu'il y a, dans de tels dialogues, à comprendre les intentions précises de l'interlocuteur (Allen et al., 2001a).

2.4.3 Traitements syntaxiques et « fautes »

Le paragraphe précédent a montré l'importance de la syntaxe dans le pouvoir d'expression du langage et donc la nécessité de sa prise en compte.

L'analyse syntaxique est l'un des domaines privilégiés du TAL. Aucune grammaire ne sait décrire complètement une langue naturelle, quelle qu'elle soit ; cependant, les formalismes récents savent décrire la plupart des phénomènes syntaxiques courants, même lorsqu'ils sont non triviaux. Ces formalismes sont parfois difficiles à implémenter (Abeillé and Blache, 2000) et ils présupposent très souvent la correction grammaticale des énoncés analysés. Or, dans les énoncés soumis à la CAP dans un système de DOHM, les erreurs de reconnaissance et les inévitables reprises et autocorrections du locuteur qui cherche ses mots rendent cette correction illusoire.

Pour pouvoir adapter cependant certains traitements à la langue orale spontanée, plusieurs stratégies sont possibles : une correction des énoncés préalable à leur analyse ou, au contraire, un traitement particulier réservé aux énoncés pour lesquels une analyse plus « classique » a échoué.

Correction des « fautes »

Dans la mesure où c'est « *par rapport à l'écrit admis comme norme qu'ont été effectuées la plupart des descriptions syntaxiques* » (Kerbrat-Orecchioni, 1990), « *la première tentative est de ramener les productions orales au format de l'écrit* » (Blanche-Benveniste, 2002). Le plus simple des traitements destiné à ramener l'analyse de l'oral à l'écrit est un traitement automatique des « fautes ».

Ces types de traitements sont contestés par les linguistes car ils sous-entendent que l'on veuille projeter les modèles de la langue écrite sur la langue parlée et ils impliquent nécessairement une modification des énoncés source. C. Blanche-Benveniste affirme que : « *en purifiant l'objet d'étude, on l'a changé* » (Blanche-Benveniste, 2002). Si l'auteur s'adresse essentiellement aux grammairiens, elle évoque également « *les informaticiens* » qui

« *prennent possession du domaine en utilisant souvent, pour analyser l'oral, des méthodes qu'ils ont rodées pour le traitement de l'écrit et qu'ils pensent pouvoir appliquer moyennant quelques adaptations mineures.* »

Elle admet cependant que cette opération est envisageable pour

« *les tâches dans lesquelles la langue est considérée comme un simple moyen de communication dont on peut transférer le contenu dans des formes diverses* ».

Le problème posé est donc de savoir si ces « changements » sont faciles à mettre en œuvre et s'ils ont une incidence ou non sur la représentation que l'on compte faire du sens de l'énoncé. Il va de soi que la représentation sémantique est d'autant plus sensible qu'elle se veut précise.

Un traitement possible consiste à essayer d'éliminer les « scories » que sont les fragments de mots, les hésitations, rectifications et répétitions (désignés comme des « réparations » par les linguistes : cf. §4.2, page 52). La détection des réparations peut être menée dès l'étape de la reconnaissance, comme le proposent P.A. Heeman et J.F. Allen : le modèle de langage utilisé pour la reconnaissance inclut les marqueurs et utilise des indices fournis par l'intonation (Heeman and Allen, 1999). Les auteurs arrivent ainsi à « corriger » les 2/3 des réparations avec 74% de précision. Ce résultat montre l'importance du nombre de fausses erreurs détectées et fait douter de l'intérêt d'un tel traitement.

La « correction » des réparations peut aussi être envisagée comme un stade préalable à l'analyse. Ainsi, J. Bear décrit une détection automatique des réparations basée sur des critères syntaxiques et morphologiques de surface : la reconnaissance de « motifs » (Bear et al., 1992). Cette détection est surgénérative ; elle confond les « véritables » réparations avec d'autres phénomènes linguistiques (cf. §4.2). Ainsi, dans l'énoncé « *show the flight time flight date* » (*faites voir les horaires des vols les dates des vols*), la répétition du mot *flight* provoque l'identification du motif $M_1R_1M_1R_1$ (où M correspond au mot répété et R correspond à un mot réparé) :

show the flight time flight date
 $M_1 \quad R_1 \quad | \quad M_1 \quad R_1$

La correction liée à cette fausse réparation correspond à la perte d'une partie du sens de la phrase : « *show the flight date* » (*faites voir les dates des vols*).

On ne voit pas très bien en effet comment une analyse sur les motifs peut distinguer les traitements différents à effectuer sur « *une réservation pour mardi une réservation pour demain* » suivant le jour de la semaine où cette requête est faite. Si le lendemain n'est pas un mardi, il s'agit d'une réparation ou d'une énumération alors que dans le cas contraire il s'agit très probablement d'une répétition avec enrichissement lexical. Bear et ses confrères reconnaissent qu'il faudrait ajouter à cette étude des motifs des arguments sémantiques et prosodiques. Dans le chapitre 4, on verra que le simple repérage des répétitions pose déjà quelques problèmes (cf. page 52).

Un « nettoyage » fondé sur une analyse de surface des productions orales s'avère donc délicat et son efficacité reste à prouver.

Traitements particuliers pour les énoncés « mal-formés »

Une autre approche est utilisée pour parvenir à une analyse syntaxique des énoncés oraux inspirée de celle des textes écrits en respectant la nécessité de correction grammaticale souvent impliquée par ce type d'analyse : elle consiste à utiliser un traitement de substitution pour les énoncés dits « mal-formés », pour lesquels un traitement « classique » échoue.

Le système TINA correspond à une telle approche. On peut citer également les travaux de N. Boufaden et de ses collègues ; les auteurs s'appuient sur le principe selon lequel « *la formulation de l'oral peut être transformée afin d'obtenir une forme semblable à celle de l'écrit* » (Boufaden et al., 1997).

Le système intègre un noyau qui utilise un analyseur syntaxique de l'écrit et un système périphérique qui intervient au cas où le noyau n'arrive plus à progresser dans son analyse. Ce dernier s'appuie entre autres sur la caractérisation des marques de surface étudiées par N. Colineau (Colineau, 1997).

Ces approches montrent qu'il est possible d'appliquer aux énoncés oraux certains traitements conçus pour l'analyse de textes écrits. En même temps, le principe de faire une distinction entre les « bons » énoncés auxquels le traitement s'applique et les « autres » pour lesquels il convient de changer de méthode reste un problème difficile.

2.5 Conclusion

Le problème des approches à adopter pour la compréhension en DOHM reste ouvert : elles sont actuellement extrêmement diversifiées, en particulier pour ce qui concerne les représentations

sémantiques rendues et l'utilisation de la syntaxe.

Pour implémenter des systèmes de DOHM dans des domaines plus larges, pour des tâches plus complexes ou avec une plus grande part d'initiatives laissée aux utilisateurs, l'idée qu'une amélioration de la compréhension est nécessaire commence à s'imposer. Par exemple, I. Lewin et ses collègues ont essayé de tester l'utilité d'une analyse linguistique fine des énoncés : ils ont comparé les performances d'un système de DOHM finalisé avec deux modules de compréhension mis en parallèle : le premier faisait une analyse linguistique approfondie des énoncés, le second une analyse par simple détection de mots-clés. Les résultats ont corroboré leurs hypothèses selon lesquelles une analyse sélective tend à devenir insuffisante lorsque le dialogue permet aux utilisateurs de prendre des initiatives (Lewin et al., 1999). Pour l'instant, on ignore quelles sont les meilleures approches à adopter pour une analyse plus fine des énoncés et si celles qui sont utilisées jusqu'à présent le permettent (Hirschman, 1998).

Il est d'autant moins facile de cerner le problème que l'on ne sait pas encore non plus cerner précisément le domaine d'application du DOHM lui-même. Actuellement, ces technologies langagières sont loin de faire partie des usages de tout un chacun. Pour la plupart des utilisateurs potentiels, elles ne sont au mieux qu'un objet de curiosité. Leur développement conditionne le type des usagers auxquels elles peuvent s'adresser : si elles permettent à leurs utilisateurs de s'exprimer presque « naturellement », elles peuvent devenir une interface privilégiée et susciter l'intérêt d'utilisateurs non-spécialistes, tels que les personnes âgées ou affectées de certains handicaps. Dans le cas contraire, elles peuvent rester des interfaces cantonnées dans des utilisations marginales. D'après Pierrel et Romary, les progrès passent

« par une meilleure intégration des travaux des linguistes et des philosophes du langage, par trop ignorés dans ces systèmes de première génération . »

Les linguistes étudient depuis longtemps la communication dans le dialogue humain et les formes de l'expression orale. Leurs travaux sont d'ores et déjà exploités par les informaticiens, puisqu'ils sont utilisés pour élaborer des modèles de communication dans les systèmes multi-agents (Moulin, 1998; Rousseau et al., 1996). Même si certaines observations ne correspondent pas au contexte particulier d'un dialogue Homme-Machine, leurs conclusions permettent néanmoins de mieux connaître le domaine d'étude. Les deux chapitres suivants (chapitre 3 et 4) leur sont consacrés.

Chapitre 3

Dialogue et communication

3.1 Introduction

La question n'est pas tranchée de savoir si le langage constitue un cas particulier de communication animale ou une exception humaine ; quoiqu'il en soit, « *apprendre à parler fait partie du patrimoine génétique humain* » (Christophe, 2002). Les êtres humains passeraient environ 20% de leur temps éveillé à converser et cette activité serait, selon J.-L. Dessalles, la fonction première du langage (Dessalles, 2000)¹. Le langage est utilisé par l'homme pour structurer sa pensée mais il est aussi le premier de ses moyens de communication. Les propos échangés sont accompagnés de mimiques gestuelles variées et la distinction entre données linguistiques et extra-linguistiques est sujette à discussion entre les linguistes (Gadet, 1997) : l'intonation par exemple est parfois considérée comme appartenant aux données gestuelles plutôt qu'aux données linguistiques.

La langue humaine est extrêmement complexe et ambiguë et sa richesse est sans doute liée à cette ambiguïté. La machine ne connaît quant à elle que les langages formels. La possibilité même de la Compréhension Automatique de la Parole fait l'objet d'une polémique : le mot *Compréhension* est en effet lié à celui de *sens* alors que le mot *Automatique* implique l'utilisation d'un formalisme. Certains détracteurs de l'IA arguent que « *toute formalisation du sens implique un traitement syntaxique qui ne s'intéresse qu'à la forme et entraîne la disparition du sens* », selon une formule reprise de G. Sabah (Sabah, 1997). À l'inverse, D. Vanderveken pense que « *il y a peu de différences théoriques importantes entre les langues naturelles et formelles* » (Vanderveken, 1999).

Quelle que soit l'issue de ce débat et s'il est sans doute vrai que la formalisation de la langue humaine dans toute sa complexité n'est pas réalisable, la compréhension automatique de la parole peut être avant tout conçue comme un élément d'une interface particulièrement « conviviale » entre l'homme et la machine.

Même si adapter son discours à son interlocuteur fait partie des compétences linguistiques de chacun, il n'en reste pas moins qu'actuellement, « parler » à une machine ne peut ressembler à un acte « naturel » que si cette communication s'apparente peu ou prou à un échange verbal entre êtres humains. Cette communication verbale entre humains a fait depuis longtemps l'objet d'études de la part des philosophes du langage ; les modèles élaborés pour l'analyse des échanges verbaux qui sont présentés dans la suite de ce chapitre (§3.2 et §3.3) sont devenus progressivement de plus en

1. La thèse de l'auteur est que la conversation est chez les humains le moyen essentiel d'accéder à la reconnaissance sociale.

plus complexes. Ils montrent la diversité des éléments à prendre en compte pour analyser ce type de communication et la difficulté des problèmes à résoudre pour tenter de la simuler.

Cependant, un dialogue Homme-Machine orienté par une tâche précise dans un domaine circonscrit offre un cadre très particulier qui permet de simplifier considérablement les modèles généraux de communication ; l'étude de ces simplifications fait l'objet de la dernière partie de ce chapitre (§3.4).

3.2 Pragmatique et actes de langage

Les analyses traditionnelles du langage (comme par exemple celle de Montague) reposent sur l'idée que la fonction première du langage est de décrire le monde et que la notion de signification repose sur celle de vérité. Ces analyses classiques sont accusées d'écarter

1. les sujets parlants ordinaires,
2. le contexte ordinaire des échanges verbaux,
3. les usages ordinaires de la langue.

La *pragmatique* étudie l'usage du langage dans la communication et réintroduit dans les analyses ces trois composantes. Son développement débute avec la théorie des *actes de langage* selon laquelle la principale fonction du langage est d'accomplir des actions.

3.2.1 Les actes de langage

Dans « *How to do things with words* » (Austin, 1962), Austin dénonce ce qu'il appelle l'« *illusion descriptive* » selon laquelle tout énoncé correspond à la description d'un état de choses ou à l'affirmation d'un fait. Seuls les énoncés qui répondent à ce critère (qualifiés de *constatatifs*) se prêtent à une interprétation sous la forme d'une valeur de vérité : *vrai, faux*. Il fait remarquer que beaucoup d'énoncés ne décrivent rien et que, par conséquent, ils ne sont ni vrais ni faux. Ainsi, des énoncés tels que « *je te promets que je viendrai* », « *venez* », « *quelle heure est-il ?* » servent à manifester une intention, donner un ordre ou obtenir un renseignement. Ces énoncés qualifiés de *performatifs* accomplissent une action ; Austin propose de leur attribuer les qualificatifs de *heureux* ou *malheureux* plutôt qu'une valeur de vérité, suivant que l'acte correspondant réussit ou échoue. Très rapidement, la distinction entre énoncés *performatifs* et *constatifs* s'avère difficile voire impossible et cette distinction est remplacée par une distinction entre trois actes de langage que l'on peut accomplir en parlant :

1. *L'acte locutoire* est l'acte que l'on accomplit dès que l'on dit quelque chose, indépendamment du sens que l'on veut communiquer.
2. *L'acte illocutoire* est l'acte accompli en effectuant l'acte locutoire : décrire, renseigner, questionner, avertir, menacer, etc.
3. *L'acte perlocutoire* désigne les effets produits par l'acte locutoire.

Austin tente ensuite d'établir une taxinomie des actes illocutoires, dits actes de langage. Celle-ci est ensuite reprise et complétée par Searle, qui la fait s'appuyer sur un certain nombre de critères quelques peu hétéroclites, tels que le but de l'acte, la direction d'ajustement entre les mots et le

monde, les états psychologiques exprimés (Searle, 1970). Searle dégage cinq classes essentielles d'actes :

1. Les actes *assertifs* : ce sont les assertions, affirmations etc.
2. Les actes *directifs* ou *exercitifs* : ils correspondent à des ordres, des demandes, des conseils etc.
3. Les actes *promissifs* : ce sont les promesses, offres et invitations etc.
4. Les actes *expressifs* : félicitations, remerciements etc.
5. Les actes *déclaratifs* : leur seule expression suffit à modifier l'état du monde ; ce sont les déclarations de guerre, nominations, baptême etc.

Cette taxinomie continue de provoquer des discussions entre les philosophes du langage, qui tentent de l'établir sur des bases plus logiques (Vernant, 2003).

Ces classifications complexes sont critiquées par Sperber et Wilson (Sperber and Wilson, 1989) qui font remarquer que le classement d'un énoncé est souvent difficile alors qu'il n'est pas toujours nécessaire à son interprétation. Ainsi « *il pleuvra demain* » peut être un acte de promesse, de prédiction ou de menace mais cette distinction n'est pas indispensable à son interprétation. Les deux auteurs proposent de réduire à trois classes les actes de langage :

l'acte **dire que** est attribué aux phrases déclaratives (assertions, refus, promesses, prédictions) : par exemple « *j'ai réservé une chambre* ».

l'acte **dire de** correspond aux phrases impératives (ordres, conseils) : par exemple « *réservez-moi une chambre* ».

l'acte **demander si** est attribué aux phrases interrogatives.

D. Vanderveken insiste sur l'universalité des actes de langage, qui servent de base à la définition d'une nouvelle logique, la *logique illocutoire* (Vanderveken, 2001). Il propose de représenter les actes illocutoires élémentaires sous la forme $F(P)$ où F est leur *force illocutoire* et P leur *contenu propositionnel*. Les notions de vérité et de succès peuvent s'appliquer à P alors que celles de satisfaction et de félicité s'appliquent à $F(P)$.

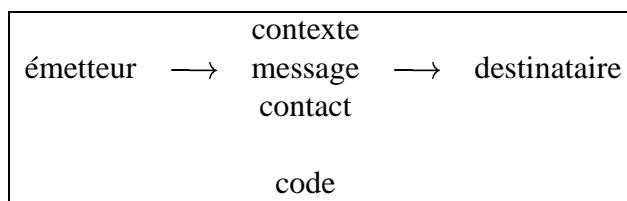
3.2.2 Interaction langagière et actes illocutoires collectifs

La théorie des actes de langage de Austin et Searle présentés précédemment se rapportent à un discours monologal ; « *elle a négligé l'analyse de la structure des échanges linguistiques* » (Vanderveken, 1999).

Le principe fondamental de l'approche interactionnelle est que « *Tout discours est une construction collective* » (Kerbrat-Orecchioni, 1990, p. 13). Elle est fondée sur les postulats suivants :

- Tout acte de parole implique l'existence d'un destinataire, et généralement, une réponse, ou tout du moins une réaction de ce destinataire.
- Dans le déroulement d'un échange communicatif, les différents participants exercent une interaction l'un sur l'autre. Ils échangent, et, ce faisant, ils modifient à la fois la perception qu'ils ont de l'autre, leurs idées et leurs comportements.

L'approche interactionnelle conduit à la définition d'unités de langage d'ordre supérieur et des actes correspondants : échanges de nouvelles, discussions, etc. ; C. Kerbrat-Orecchioni les désigne

FIG. 3.1 – *Le modèle du code*

sous le terme de *macro-actes* ou *actes conversationnels*² (Kerbrat-Orecchioni, 2002). D. Vanderveken nomme « *actes illocutoires collectifs* » (Vanderveken, 2001) les actes de conversation « joués à plusieurs » que constituent les échanges de salutation, les délibérations, les débats, etc. Il propose une logique du discours pour analyser « *la structure logique et les conditions de succès des conversations que les locuteurs humains sont capables de tenir en vertu de leur compétence linguistique* » (Vanderveken, 1999).

3.3 Modèles de communication verbale

Les études sur la communication verbale permettent d’avoir une meilleure connaissance de sa complexité. Cette complexité croissante se traduit par celle des modèles proposés.

Ainsi, dans le modèle dit *du code*, la production et la compréhension des énoncés sont considérées comme faisant appel à une compétence essentiellement voire exclusivement linguistique. Dénoncée rapidement comme trop restrictive, la notion de compétence est ensuite élargie pour devenir « *un dispositif complexe d’aptitudes où les savoirs linguistiques et les savoirs socio-culturels sont inextricablement mêlés* » (Kerbrat-Orecchioni, 1990, p. 31). Le *modèle inférentiel* élargit la notion de compétence et fait de la compréhension un processus complexe difficilement réductible à un simple décodage. Dans le mécanisme des *interactions verbales*, interviennent également les modifications et adaptations subies par les interlocuteurs lors de leurs échanges verbaux.

3.3.1 Le modèle du code

Le modèle le plus simple de la communication langagière est le modèle de Jakobson dit *du code* (Jakobson, 1963) résumé dans la figure 3.1.

Suivant ce modèle, l’émetteur encode le message qu’il veut communiquer au moyen d’une expression langagière. Le destinataire décode cette expression pour identifier le sens voulu. Deux principes fondamentaux sous-tendent ce modèle :

- Un code permet d’associer à tout sens une expression et à toute expression correspond un sens et un seul.
- Les langues humaines sont des codes.

Ainsi, la communication langagière exige la connaissance du code, assimilée à une certaine compétence³. Cette compétence est de nature linguistique : elle correspond à la connaissance des règles

2. C. Kerbrat-Orecchioni définit, en plus des actes « classiques » d’Austin et Searle des micro-actes et des macro-actes.

3. Ce terme de *compétence* est utilisé par Chomsky.

qui régissent la production et la compréhension des énoncés. Elle est partagée par tous les membres de la communauté linguistique, cette dernière affirmation étant évidemment un principe plutôt qu'une réalité. Les différences entre locuteurs sur la correspondance forme-sens pour un mot particulier sont loin d'être rares. Outre ces discordances sur le sens dit référentiel, des divergences se manifestent également dans les connotations véhiculées par les mots (Baylon and Mignot, 1995).

Malgré les restrictions précédentes, ce modèle offre un cadre particulièrement bien adapté à la compréhension automatique de la parole : il « suffit » d'apprendre le code à la machine pour récupérer dans son intégralité le message exprimé par le locuteur. Toute la difficulté réside dans la définition des règles linguistiques.

3.3.2 Le modèle inférentiel

Le modèle du code se révèle très vite notoirement insuffisant dès qu'il s'agit de représenter la communication dans un dialogue, fût-il très élémentaire. En effet, le destinataire d'un message doit faire appel à bien d'autres compétences que linguistiques pour en déterminer l'interprétation.

Sperber (Sperber, 2000) donne comme exemple la réponse « *j'ai déjà mangé* » donnée par Marie à Pierre qui lui a demandé si elle voulait dîner avec lui. Au sens littéral le plus strict, ce que dit Marie signifie simplement qu'il lui est déjà arrivé de manger. Or, le message qu'elle veut transmettre à Pierre est beaucoup plus fort. Elle veut lui faire comprendre qu'elle vient de manger le soir même et que, par conséquent, elle refuse son invitation. Selon une expression empruntée à Sperber, le sens linguistique « *sous-détermine le vouloir-dire du locuteur* », en ce sens qu'il ne constitue qu'une indication incomplète de ce vouloir-dire. Ainsi, dans les communications humaines, l'*implicite* est constant et « *tout discours s'échafaude sur la base de « postulats silencieux » engrangés dans la compétence encyclopédique* » (Kerbrat-Orecchioni, 1986).

Le modèle *inférentiel* correspond à un modèle de la communication destiné à prendre en compte cette sous-détermination (Kerbrat-Orecchioni, 1980). Le locuteur utilise pour coder son message des *compétences* linguistiques, mais aussi para-linguistiques (comportements, gestes, mimiques), idéologiques et culturelles, psychologiques, etc. Ces compétences permettent de savoir que dire et comment le dire, à qui et dans quelles circonstances. Ainsi, le locuteur ne produit pas un encodage du sens qu'il veut communiquer, mais seulement un indice de ce sens. Quant au récepteur, il *infère* ce que le locuteur a voulu dire à partir de ce qu'il a réellement dit, en utilisant des compétences du même ordre. Le terme d'*inférence* renvoie à un mécanisme proche de celui du raisonnement mais de nature largement automatique et inconsciente. Cette capacité d'inférence excède la seule compétence linguistique.

Deux théories sont en concurrence pour expliquer comment fonctionne le mécanisme d'inférence (Moeschler, 2000) :

- **Les implicatures** : selon Grice (Grice, 1979), le mécanisme d'inférence qui fait appel, bien sûr, à des informations contextuelles, repose sur un « *principe de coopération* ». Le récepteur suppose que le locuteur adapte son comportement pour un bon déroulement de la conversation et qu'il applique certaines règles dites « *maximes conversationnelles* » :
 1. *Maxime de quantité* : être aussi précis que possible mais pas plus que nécessaire.
 2. *Maxime de qualité* : n'affirmer que ce que l'on tient pour vrai.
 3. *Maxime de relation* : être pertinent par rapport au sujet de la conversation et à ce que l'on veut communiquer.

4. *Maxime de manière* : être le plus clair possible. Le discours doit être ordonné, clair et précis.

Le respect de ces différentes maximes correspond à une certaine compétence, variable selon les individus et qui excède la seule compétence linguistique. Leur non-respect correspond soit à une absence de compétence, soit, lorsqu'il est délibéré, à un effet de style (ironie, par exemple). Le résultat de l'application de ces maximes au message verbal reçu par le récepteur conduit à des « *implicatures conversationnelles* ». Par exemple, de l'énoncé « *je veux réserver pour quatre jours* » et en appliquant la maxime de quantité, on peut inférer que le locuteur veut réserver pour exactement quatre jours et non pas trois même si l'énoncé « *je veux réserver pour trois jours* » est une conséquence sémantique du précédent au sens de la logique classique.

Grice définit également des « *implicatures conventionnelles* » qui diffèrent des précédentes en ce qu'elles sont déclenchées non pas par l'application des maximes précédentes, mais par les formes linguistiques elles-mêmes. Ainsi, de l'énoncé « *je n'ai pas réussi à réserver* », on peut inférer que :

1. Le locuteur n'a pas réservé.
2. Le locuteur a essayé de réserver.

L'un des mérites de cette théorie est d'expliquer certaines variations de sens dans des expressions linguistiques comme étant le résultat de principes pragmatiques généraux. Par exemple, peuvent s'expliquer de cette manière certains usages du *ou* en tant que *ou exclusif*. Ainsi, l'expression « *fromage ou dessert* » correspond à un *ou exclusif* par le résultat d'une implicature quantitative : si le client avait droit à fromage et dessert, l'information communiquée serait « *fromage et dessert* » par application de la maxime de quantité. L'expression « *fromage ou dessert* » communique donc la double information *il est vrai que fromage ou dessert* et *il est faux que fromage et dessert* ; cette interprétation correspond bien à la définition du *ou exclusif*.

- **La pertinence** : suivant la théorie plus récente de la pertinence élaborée par Sperber et Wilson (Sperber and Wilson, 1989), le récepteur élabore une hypothèse sur le sens voulu par le locuteur en étant guidé par une « *pertinence* » optimale de l'énoncé dans les contextes particuliers où il est produit. Un énoncé est considéré comme d'autant plus pertinent qu'il produit un maximum d'*effets contextuels* pour un minimum d'*efforts cognitifs*.

Les *efforts cognitifs* désignent l'effort demandé au récepteur pour comprendre l'information.

Les *effets contextuels* correspondent aux effets produits par cette information.

La pertinence d'un énoncé mesure donc à la fois la facilité de compréhension du message et son intérêt.

D'après Sperber (Sperber, 2000), la compréhension inférentielle est rendue possible par une compétence spécifique, celle de « *la possibilité de se représenter mentalement les représentations mentales d'autrui* ». Elle est largement dépendante des individus.

La liste des compétences nécessaires à la communication varient suivant les auteurs et la seule qui est universellement reconnue est la *compétence linguistique*. Tous s'accordent cependant à reconnaître que la compréhension entre interlocuteurs fait appel à des compétences d'ordres variés. C. Kerbrat-Orecchioni tente d'explicitier les qualités qu'elle désigne comme étant « nécessaires

aux sujets parlants » (Kerbrat-Orecchioni, 1986). Outre les compétences linguistiques et encyclopédiques, la compétence logique joue, suivant cet auteur, un rôle très important dans la compréhension ; elle permet en particulier d'extraire de nombreux présupposés et sous-entendus. Par exemple, c'est la logique classique qui permet de déduire de l'énoncé « *frais, le café ne brille pas* » la première conclusion « *s'il brille, c'est qu'il n'est pas frais* », par contraposition. En revanche, c'est la logique « naturelle » des sous-entendus qui permet d'en déduire qu'il peut arriver que le café brille et que « *s'il n'est pas frais, il brille* », ces deux conclusions sous-entendues permettant de faire de la brillance du café un moyen de reconnaître son état de fraîcheur. Cet exemple peut également être cité pour illustrer une inférence dû au principe de pertinence selon Sperber ; l'énoncé « *frais, le café ne brille pas* » est en effet beaucoup plus pertinent si l'on peut en déduire les deux sous-entendus. Enfin, la compétence dit « rhétorico-pragmatique » est la dernière des compétences nécessaires citées par C. Kerbrat-Orecchioni ; elle correspond au respect des maximes de Grice, respect plus ou moins strict mais néanmoins toujours présent chez les locuteurs sensés.

Par ailleurs, l'approche interactionnelle fait valoir que les influences que les interlocuteurs exercent l'un sur l'autre modifient leurs compétences respectives au cours de l'échange. Ceci permet de dire à C. Kerbrat-Orecchioni que « *la clé qui permet d'effectuer les opérations d'encodage et décodage ... est en partie construite au cours de l'interaction* » (Kerbrat-Orecchioni, 1990, p. 28).

Quels que soient les mécanismes de fonctionnement de l'inférence, la compréhension telle qu'elle est conçue dans le modèle inférentiel est un processus très complexe. Elle fait appel à un faisceau de compétences variées et diverses, difficiles à cerner et donc a fortiori à modéliser.

Elle est également un processus à très haut risque, soumis à des erreurs. La compréhension partielle, voire les erreurs de compréhension ne sont pas considérées comme des exceptions ou des accidents, mais comme des phénomènes fréquents qu'il convient donc de décrire et d'analyser. Ce dernier point, est, nous semble-t-il, essentiel.

3.4 Communication et dialogue Homme/Machine finalisé

Les modèles précédents se réfèrent à des communications entre locuteurs humains. Leur complexité ne laisse que peu d'espoir sur la possibilité de pouvoir, à court ou moyen terme, automatiser la compréhension des échanges verbaux dans un cadre qui ne soit pas très fortement contraint.

Dans un dialogue Homme/Machine finalisé, l'utilisateur « parle avec » la machine dans le but d'accomplir une tâche précise dans un domaine délimité. Une telle communication présente évidemment des particularités par rapport aux modèles de conversation très généraux de dialogue Homme/Homme : le rôle de l'interaction y est relativement réduit et certains contextes sont bien déterminés. Ces deux facteurs permettent de penser que la compréhension automatique est possible dans un tel cadre à condition d'accepter, comme dans les communication Homme/Homme, la présence d'erreurs de compréhension et la nécessité de leur prise en compte dans la gestion du dialogue.

L'objet de cette section est d'essayer de dégager quelques principes à mettre en œuvre pour la compréhension des énoncés dans de tels systèmes, compte tenu de ces modèles et de ces particularités. Il s'agit aussi d'essayer de définir ce que peut être recouvrir alors le terme de « compréhension ».

3.4.1 Dialogue Homme/Machine et interaction

Dans un dialogue Homme/Machine et plus particulièrement dans le cas d'un serveur vocal, des éléments de la communication entre humains telle qu'elle a été décrite par les philosophes du langage sont absents ou négligés, ce qui permet de diminuer la complexité de l'analyse. Cependant, cette simplification induit l'absence de certains indices : elle peut donc également être considérée comme un manque d'informations qui auraient été susceptibles de guider la compréhension.

- **Indices gestuels et para-linguistiques :** dans un serveur vocal, les indices gestuels et comportementaux que peut émettre l'utilisateur humain sont délibérément ignorés. On peut considérer que ne pas intégrer ce type d'indices dans un message qu'il veut transmettre fait partie de la compétence de l'utilisateur.

Le message linguistique proprement dit est le plus souvent simplifié puisque l'énoncé source transmis par le module de reconnaissance au module de dialogue sous la forme d'un ensemble ordonné de mots ne comporte que rarement des indications de pause et de prosodie. Négliger ce type d'indices est certainement se priver d'une source importante d'informations. À condition de savoir les utiliser, ces marques pourraient certainement apporter des précisions intéressantes pour l'analyse de l'énoncé. Il suffit pour s'en convaincre de penser aux marques de l'interrogation, de plus en plus souvent supportées, dans la langue française orale, par la seule intonation. Le TAL s'intéresse à une intégration de la prosodie comme composant de l'analyse linguistique comme en témoignent certains travaux relativement récents sur ces sujets mais cette intégration est loin d'être simple (cf. §1.5.2, page 17).

- **Respect des tours de parole :** C. Kerbrat-Orecchioni fait observer que dans un dialogue ordinaire, les énoncés des différents interlocuteurs se chevauchent et s'entremêlent, ce qui ajoute encore à la complexité des mécanismes de l'interaction (Kerbrat-Orecchioni, 1990). Dans un dialogue Homme/Machine, on peut supposer que le dialogue respecte le processus des « tours de parole ». Le récepteur, passif attend la fin du message du locuteur et l'on part du principe que l'utilisateur du système n'interrompera pas les réponses générés par ce dernier. Cependant, là aussi, des efforts commencent à être faits pour casser la rigidité très peu naturelle induite par une séquentialité trop rigide des tours de parole (cf. 1.4.4).
- **Faible subjectivité :** l'un des problèmes de la compréhension mis en avant par C. Kerbrat-Orecchioni est celui de la subjectivité (Kerbrat-Orecchioni, 1980). L'auteur montre que cette subjectivité est partout présente dans le langage. Le fait que les applications qui intéressent la compréhension correspondent à des tâches précises concernant des objets concrets réduit considérablement le problème. Il n'est pourtant pas totalement absent. Dans le domaine de l'information touristique, des qualificatifs tels que « *pas trop cher* » ou « *pas trop loin* » ont certainement un sens différent selon la nature des objets auxquels ils sont appliqués ou selon les locuteurs, et autorisent certains choix quant à l'interprétation qu'on doit leur donner mais ce problème en soulève un autre évoqué dans le point suivant.
- **Interactions et modèles de l'interlocuteur :** un dialogue est considéré comme une communication interactive entre les locuteurs, qui modifient au fur et à mesure de son déroulement les images qu'ils ont de leurs interlocuteurs. Sans nul doute, l'utilisateur humain d'un dialogue de DOHM modifie l'image qu'il a du système lorsqu'il l'utilise. S. Rosset rapporte les résultats d'une expérience faite sur les utilisateurs du système ARISE (Rosset, 2000) : les

images qu'ils disent avoir du système après leurs premier et deuxième appel différent sensiblement. D'une manière d'ailleurs assez paradoxale, ils affirment une satisfaction moindre alors même qu'ils ont plus souvent obtenu le renseignement qu'ils ont demandé (probablement grâce à une meilleure connaissance pragmatique).

En sens inverse, les stratégies de dialogue intègrent parfois des images des utilisateurs sous la forme de novices, occasionnels ou experts (Rosset, 2000). Un expert se verra par exemple proposer un dialogue moins directif et permettant d'aller plus directement au but qu'un novice. Même si ces modèles sont encore très rudimentaires, ils constituent néanmoins un premier pas dans la prise en compte de l'interlocuteur.

Par ailleurs, dans presque toutes les applications, le principe de sincérité de l'utilisateur est admis. Cependant, là encore, cette règle connaît quelques exceptions : par exemple pour certaines opérations bancaires, détecter un utilisateur frauduleux peut constituer un enjeu important.

- **Actes illocutoires** : certains actes illocutoires n'ont guère leur place en dialogue Homme-Machine. On peut exclure d'emblée les actes *promissifs* : il semble en effet difficile ou pour le moins inapproprié de proférer des promesses ou des menaces à l'endroit d'une machine. Quant aux actes *expressifs*, ils ne sont guère représentés que sous la forme de remerciements ; ces derniers sont une manifestation de satisfaction de la part de l'utilisateur, qui confirme un bon déroulement du dialogue. Si l'on s'en tient à la taxinomie de Searle (cf. page 41), il reste donc essentiellement des actes *assertifs*, *directifs* et *déclaratifs*.

Par ailleurs, les *actes illocutoires collectifs* correspondent dans une large mesure à la tâche que se propose d'accomplir l'homme avec l'aide de la machine. Cet acte est donc a priori connu, ou du moins il appartient à une liste de tâches connues : celles pour lesquelles la machine a été conçue (on peut décider de ne pas avoir à prendre en compte les utilisateurs dont l'objectif premier serait de jouer avec la machine, pour tester ses capacités, même si l'on connaît leur existence).

3.4.2 Compréhension partielle et erreurs de compréhension

Les modèles récents de la communication Homme/Homme montrent qu'il s'agit d'un processus « à haut risque ». Néanmoins, après des échanges verbaux plus ou moins longs, la plupart des communications verbales entre humains aboutissent et les interlocuteurs finissent par se faire comprendre. Pour la mise en œuvre d'un dialogue Homme/Machine, on peut retenir l'idée essentielle que la compréhension partielle d'un message ou même les erreurs de compréhension sont des aléas ordinaires et qu'ils ne doivent pas provoquer l'échec du dialogue. Pour le module de compréhension, ce principe implique deux conséquences essentielles :

1. La compréhension d'un énoncé peut n'être que partielle. L'analyse doit pouvoir traiter les énoncés incomplets et aussi pouvoir laisser de côté éventuellement des bribes d'énoncés sémantiquement incompatibles avec le reste de cet énoncé.
2. L'analyse doit pouvoir reconnaître des actes de langage tels que des refus, des corrections et des complétions des requêtes.

3.4.3 Définitions de la compréhension en DOHM

Il n'est pas évident de déterminer ce que recouvre exactement le terme de *représentation sémantique* d'un énoncé. Les linguistes distinguent la « *forme logique* » de la « *forme propositionnelle* ». La « *forme logique* » correspond à la représentation sémantique que l'on peut construire à l'aide des informations linguistiques (donc en se fondant essentiellement sur des indications syntaxiques) et la « *forme propositionnelle* » à la représentation pragmatique obtenue en enrichissant la forme logique d'indications contextuelles (Moeschler, 2000). Classiquement, en TALN, on distingue l'interprétation dite *littérale* de l'interprétation *contextuelle* ; ces deux dénominations correspondent à peu près aux formes logique et propositionnelle des linguistes. Dans un système de Compréhension Automatique de la Parole (CAP), cette distinction peut sembler discutable.

Actuellement en effet, la Compréhension Automatique de la Parole ne s'exerce que pour des applications particulières, dans des domaines finalisés. Le but du dialogue est d'obtenir des informations liées à une base de données ou de faire exécuter une tâche donnée (cf. chapitre 1). L'application se place donc dans un domaine particulier (un monde d'objets en nombre restreint et sur lesquels on peut définir une certaine connaissance) et se réfère à une tâche spécifique (ordres précis à exécuter ou demandes d'informations). Ce caractère finalisé des applications permet ainsi que l'on puisse tirer parti d'informations sémantiques liées à un micro-monde bien déterminé. Or, le caractère spontané de l'expression orale et les erreurs de reconnaissance produisent un message « bruité ». Comme une analyse syntaxique complète des énoncés est illusoire, l'analyse s'appuie toujours sur le caractère finalisé de l'application, et donc sur le contexte particulier de ce micro-monde. Dans ces conditions, « l'interprétation littérale » n'est pas la « *forme logique* » des linguistes ; elle contient des éléments qui relèvent de la pragmatique. Le contexte auquel fait référence le terme d'« *interprétation contextuelle* » se limite alors au seul contexte du dialogue.

Dans ces conditions, on peut comprendre « l'interprétation contextuelle » comme une utilisation d'éléments du contexte de dialogue qui permette plusieurs niveaux de traitements tels que :

- la résolution des références,
- la reconnaissance des intentions du locuteur par rapport au contexte du dialogue.

Cependant, se priver du contexte de dialogue pour faire une « *interprétation littérale* » présente certains risques. En effet, même en l'absence d'agrammaticalités et d'erreurs de reconnaissance, certains énoncés ne se prêtent que difficilement à une interprétation sans prise en compte du contexte de dialogue. Il est même des cas où l'absence de prise en compte de ce contexte rend les énoncés incohérents, au sens de l'interprétation littérale telle que nous avons envie de la définir : dans l'énoncé « *deux pour demain* » par exemple, aucun lien sémantique ne relie a priori les deux propriétés⁴. Par contre, son interprétation ne pose aucun problème si l'objet du contexte est une réservation de places de spectacle. Il semble bien que ces difficultés à distinguer ces différents niveaux soient dues à la complexité de la compréhension telle que la décrit le *modèle inférentiel*. Le sens d'un énoncé est inféré (*il se déduit inconsciemment*) à partir d'un réseau d'indications (*d'hypothèses implicites*) auxquelles appartient le contexte de dialogue.

Plusieurs solutions sont possibles, non totalement exclusives les unes des autres :

1. On peut considérer qu'un excès de compétences et d'éléments d'information ne saurait être inconvenient et que le module de compréhension peut et doit s'appuyer sur le contexte de dialogue pour résoudre certaines ambiguïtés.

4. Ce problème est soulevé par Pierrel et Romary dans (Pierrel and Romary, 2000).

2. En présence d'ambiguïtés, le résultat de l'analyse peut proposer en résultat plusieurs solutions.
3. L'interprétation littérale peut correspondre à une analyse incomplète et *non destructive* : elle consiste alors en une étude des dépendances entre constituants faite avec les éléments d'information disponibles, avec un minimum de risques et en respectant les éléments présents dans l'énoncé (en évitant les opérations de « nettoyages » susceptibles d'éliminer des éléments signifiants).

Si les deux dernières solutions et en particulier la dernière sont parfois praticables pour certaines applications pour lesquelles des représentations sémantiques sous-spécifiées peuvent suffire (une levée totale de l'ambiguïté n'est pas toujours nécessaire en traduction car certaines langues permettent une traduction directe des ambiguïtés) (Danlos, 2003), elles ne le sont généralement pas dans un système de dialogue où l'on a besoin de savoir exactement ce que désire l'utilisateur. Elles consistent à repousser la difficulté à un stade ultérieur et à laisser au module de dialogue le soin de gérer les ambiguïtés. Le problème est alors ramené à celui de la répartition des tâches entre le module qui assure la compréhension et celui qui assure la gestion du dialogue.

Dans certains systèmes (Rosset, 2000), l'interprétation littérale est réduite à un simple étiquetage syntaxico-sémantique des segments conceptuels contenus dans l'énoncé. C'est le module de dialogue qui « infère » le sens de l'énoncé à partir de ces éléments et du contexte et il lui revient aussi de corriger éventuellement l'interprétation littérale ; ce module assure donc une partie importante de la compréhension.

À l'inverse, on peut concevoir que le module de compréhension dispose du contexte du dialogue et se charge de lever les ambiguïtés. La tâche du module de dialogue consiste alors à assurer l'interface avec la base de données d'une part et à gérer le dialogue d'autre part, en choisissant la nature des réponses à donner à l'utilisateur et les éventuelles questions à lui poser.

Principe de pertinence

Quelle que soit la séparation exacte des tâches entre les modules de compréhension et de dialogue, il semble raisonnable de définir et d'adopter le principe de pertinence pour les représentations sémantiques transmises par le premier module au second : la représentation sémantique de l'énoncé doit fournir un maximum des informations « utiles » et il revient au module de dialogue d'effectuer certains choix en appliquant ce principe.

Par exemple, dans le cas où le module de compréhension ne dispose pas du contexte de dialogue, il lui est impossible de savoir si un énoncé tel que « *il a deux étoiles* » est une information ou une interrogation puisque c'est en général l'intonation qui porte la marque linguistique de l'interrogation (cf. 4.4). Le module de compréhension transmet une représentation qui ne fait pas apparaître (pour cause) la marque de l'interrogation et le module de dialogue doit éventuellement l'inférer à partir des énoncés précédemment échangés. Un autre exemple peut être donné concernant la multiplicité des interprétations possibles de certaines modalités ; ainsi, dans l'énoncé « *il peut venir* », la modalité peut être interprétée comme une permission, une capacité, une éventualité ou une possibilité matérielle (Querler, 1996).

3.5 Conclusion

Le dialogue Homme-Machine finalisé offre un contexte suffisamment précis et délimité pour que les ambiguïtés inhérentes à la communication orale soient considérablement réduites par rapport à celles normalement présentes dans un échange « ordinaire ». Cependant et même dans ce cadre, la compréhension reste un processus « à haut risque » ; pour détecter (inférer) les intentions du locuteur, il convient que l'analyse soit suffisamment précise et que le langage cible permette d'exprimer une certaine complexité. Malgré tout, il faut s'attendre à des « ratés » de la compréhension, qu'il faut contrôler et gérer ; en particulier, les confirmations et les corrections apportées par le locuteur doivent pouvoir être correctement interprétées. La palette des actes de langage à envisager doit donc être assez étendue et leur détection nécessite une analyse des énoncés aussi fine que possible.

Chapitre 4

Les spécificités de l'oral spontané

4.1 Introduction

La mise en œuvre de traitements de la langue orale spontanée efficaces passe par une bonne connaissance du matériau à analyser : le module de compréhension reçoit un message orthographique qui correspond à un ensemble ordonné de mots. Celui-ci est la traduction faite par le module de reconnaissance d'un message vocal émis par l'utilisateur du système. La reconnaissance de la parole et les erreurs qu'elle engendre ont déjà été étudiés (cf. page 7). Ce chapitre décrit les spécificités de l'expression orale spontanée et pose le problème de la connaissance que l'on peut avoir des modes d'expression effectifs des utilisateurs de tels systèmes.

L'utilisateur sait qu'il s'adresse à une machine ; néanmoins, il est censé lui parler « naturellement », comme s'il s'adressait à un interlocuteur humain. Les travaux faits par les linguistes sur les énoncés oraux décrivent les spécificités de la parole spontanée. Il convient d'étudier ce qu'ils permettent de savoir sur les énoncés à analyser, en tenant compte du fait qu'ils portent essentiellement sur des corpus de dialogue homme/homme.

Pour les linguistes, les communications orales ne sont devenues un objet d'études à part entière qu'assez tardivement, l'« écrit » ayant longtemps été considéré comme la seule référence possible. Dans cette perspective, les énoncés oraux n'étaient que des « écrits dégradés », et le « *discours oral* » « *un sous-produit du langage* » (Kerbrat-Orecchioni, 1990, p. 40). De plus, les études sur la langue orale ont été longtemps freinées par l'absence de données importantes sur lesquelles travailler. Les corpus de langue parlée étaient en effet composés d'énoncés brefs (énoncés parlés attestés d'abord, enregistrements sur magnétophones par la suite). Les très grands corpus ne sont apparus que très récemment, avec les données informatisées (Blanche-Benveniste, 2002).

C. Kerbrat-Orecchioni souligne que les descriptions syntaxiques ont été élaborées par rapport à la norme de l'écrit. Par conséquent, des notions fondamentales telle que celle de phrase sont conçues pour l'analyse des textes écrits et l'on constate qu'elles s'appliquent mal au discours oral.

On peut distinguer trois types de spécificités des énoncés oraux spontanés :

- Les « *marques du travail de formulation* » sont les « ratures » laissées par la recherche des mots.
- Les « *marqueurs du discours* » sont les petits mots qui ponctuent les énoncés oraux. Ils sont souvent les marques de l'interactivité de la communication.
- La langue parlée est caractérisée par sa diversité : sa « *variation* ». Elle se manifeste dans

toutes les composantes de la langue, de la phonétique à la syntaxe. La multiplicité des tournures possibles pour exprimer quelque chose peut donner l'impression d'un déficit de syntaxe des énoncés oraux.

4.2 Les marques du travail de formulation

Chacun, en parlant cherche ses mots et énumère parfois plusieurs solutions avant d'énoncer celle dont il pense qu'elle est la bonne ; les linguistes appellent cette recherche le « *travail de formulation* ». Les énoncés écrits représentent des « produits finis » où les marques de la recherche des mots ont été soigneusement éliminées. Par contre, comme à l'oral il est impossible d'effacer ce que l'on vient de dire, comme « *on ne peut à l'oral effacer qu'en ajoutant* » (Kerbrat-Orecchioni, 1990), les énoncés oraux comportent des « ratures »¹. Si celles-ci posent rarement problème à la compréhension dans une véritable conversation, il en va tout autrement pour l'analyse d'un énoncé sous la forme d'une liste de mots orthographiés. Elles peuvent être de natures diverses comme le montrent ces quelques exemples extraits du corpus OTG recueilli à l'office de tourisme de Grenoble (Antoine et al., 2002b) :

Répétitions :

« il y a il y a un site Air Inter ici »

« s'il vous plaît des des invitations »

Autocorrections (de portée plus ou moins grande) :

« le l'adresse de des différentes MJC »

« j'aurais voulu bien voulu un plan comme ça »

« il faut qu'on passera il faudra qu'on passe à la bibliothèque »

Inachèvements :

« le programme de l'auditorium de parce qu'il y a une grande affiche là »

Les « ratures » n'ont été que relativement peu étudiées en français. L'essentiel des travaux sur ce sujet ont été faits par l'équipe du G.A.R.S. (actuellement rebaptisée DELIC) à Aix.

L'exemple pourtant apparemment simple des répétitions étudiées par S. Henry (Henry, 2002) montre la complexité du problème. Une analyse de surface peut repérer les mots et les séquences identiques qui se succèdent ; mais ces phénomènes peuvent correspondre à l'application de règles linguistiques (« *nous nous sommes vus* ») ou à une marque d'insistance (« *nous nous avons pris le train* »). S. Henry propose le terme de « répétitions de compétence » pour ces formes de répétition par opposition aux répétitions dues au travail de formulation. Cet exemple permet déjà de comprendre pourquoi, comme on l'a vu dans le chapitre 2, les traitements pour essayer de « corriger » d'une manière plus ou moins automatique les « ratures » ne sont pas faciles à mettre en œuvre.

Le terme générique de « *réparation* » sert à désigner les phénomènes de répétitions, reprises, hésitations et autocorrections du locuteur qui cherche ses mots. La terminologie adoptée pour leur description est souvent proche de celle adoptée par Shriberg (Shriberg, 1994) ; elle correspond à un découpage de type RM-IP-IM-RR où

- RM désigne le « *reparandum* » : partie ou totalité de la séquence reprise.
- IP est le « *point d'interruption* » ; il correspond à la fin du *reparandum*.

1. F. Gadet oppose la linéarité spatiale de l'écrit à la linéarité temporelle de l'oral (Gadet, 1997).

forme dans 70% des cas. Cependant, on peut aussi observer une réparation sur le seul déterminant, à la condition qu'elle soit opérée avec un déterminant de même type (« *dans ce cet hôtel* »).

Dans la mesure où il est « corrigé », le reparandum peut être considéré comme une ébauche, une « *bribe* » inachevée. Pour extraire le sens de l'énoncé, l'important est de savoir si cette bribe possède ou non une valeur sémantique.

- **Les bribes sans intérêt sémantique** : les amorces d'un syntagme inachevé, répété ou corrigé créent des mots grammaticaux non rattachés qui peuvent être négligés :

« *pour ce dans cet hôtel* » = « *dans cet hôtel* »

« *dans ce cet hôtel* » = « *dans cet hôtel* ».

- **Les bribes qui présentent un intérêt sémantique** : certaines reformulations en strates successives ou les multidésignations correspondent à des enrichissements sémantiques du discours :

il repart *avec sa femme*

enfin *avec sa fiancée*

un hôtel *pas trop cher*

enfin *une ou deux étoiles quoi*

Dans ces deux exemples, les parties réparées précisent le sens du reparandum sans pour autant chercher à l'effacer sémantiquement. Ainsi, dans le deuxième exemple, la propriété recherchée est essentiellement le tarif modéré de l'hôtel ; le nombre d'étoiles vient comme une illustration de la demande, une fourchette de prix attendue qui vient en contrepoint de la « subjectivité » (cf. 3.4.1) contenue dans le « *pas trop cher* ».

4.3 Les marqueurs de l'interactivité

Elles incluent les marques d'hésitation mais aussi les phatiques et les régulateurs dont l'importance a été soulignée pour la communication entre les différents acteurs de la conversation (cf. chapitre 3). A priori, il semble que l'on puisse les négliger dans une analyse qui cherche à construire une représentation sémantique de l'énoncé. Cependant, cette affirmation est à nuancer pour trois raisons différentes :

1. D'abord, dans le cadre de l'énoncé lui-même, les marques d'hésitation par exemple peuvent parfois guider l'analyse en indiquant éventuellement où se situe le début du reparandum de la réparation.
2. Dans un dialogue, les marques de surface aident à appréhender l'organisation séquentielle des énoncés ; Moeschler leur attribue un rôle de *connecteur pragmatique* avec la définition suivante (Moeschler, 1995) :

« *Un connecteur pragmatique donne de l'information sur le sens du segment du discours qu'il introduit et sur la façon dont il faut comprendre la connexion.* »

N. Colineau distingue ainsi différents types d'enchaînements et les marques qui leur sont en général associées : ouverture, clôture, continuation, rupture de dialogue etc. (Colineau, 1997).

3. Certains mots ou expressions sont insérés un peu n'importe où dans les énoncés, sans que les locuteurs en aient nécessairement conscience et sans être porteurs d'une signification bien définie ; c'est le cas par exemple de la suite « *en fait* », massivement présente dans tous les

productions langagières contemporaines (Savelli, 2001). Négliger ce type de marqueurs ne constitue pas un réel inconvénient pour une analyse qui cherche essentiellement à dégager des informations pratiques. Malheureusement, certains de ces mots tels que « *quoi* » ou « *comment* » sont polyvalents ; si une partie de leurs occurrences ne correspondent qu'à de simples marqueurs du discours, d'autres peuvent posséder un réel poids sémantique : des formes interrogatives pour les deux exemples cités. Des études linguistiques telles que celle de C. Chanet sur la particule « *quoi* » peuvent alors apporter une aide précieuse pour aider à lever l'ambiguïté (Chanet, 2001).

Cependant, comme toutes les manifestations de l'interaction, il est à prévoir que ces marqueurs doivent être moins présents dans le discours d'un humain s'il sait qu'il s'adresse à une machine.

4.4 Variation et syntaxe de l'oral

Dans *Le Français ordinaire* (Gadet, 1997), F. Gadet étudie comment la langue parlée peut varier au sein d'une même communauté linguistique. Les variations s'observent entre les différents locuteurs en fonction des régions, des classes sociales, de l'âge, des époques, etc. Elles concernent la phonologie, au travers par exemple des accents régionaux : prononciation des voyelles, liaisons, *e* muet etc. et peuvent donc poser des problèmes au module de reconnaissance de la parole. Elles peuvent également se manifester entre les productions d'un même locuteur en fonction de sa situation : le style d'un individu varie suivant les personnes auxquelles il s'adresse et les enjeux sociaux engagés dans ses échanges verbaux.

Ces variations concernent également la construction de l'énoncé et rejoignent le problème de la « *syntaxe* » de l'oral. Ainsi par exemple, F. Gadet souligne l'extrême diversité que peuvent prendre à l'oral les formes interrogatives et cite en exemple plus de quinze formes différentes possibles pour un énoncé aussi simple que « *Quand est-il venu ?* ». En particulier, la marque de l'interrogation est très fréquemment portée par la seule intonation, sans recours aux mots interrogatifs. Une telle forme interrogative est qualifiée de *forme totale* en opposition aux *formes partielles* où subsiste au moins un mot interrogatif. D'après des études sur corpus de l'auteur, la proportion des énoncés interrogatifs totaux peut atteindre 95% de ces énoncés. Bien sûr, ces résultats doivent être pris en compte pour la compréhension d'énoncés émis dans un contexte de dialogue dont le but final est d'interroger une base de données.

L'expression orale, plus souple que l'expression écrite, permet donc une multiplicité de tournures possibles ; cette extrême variabilité accrédite l'idée d'une syntaxe déficiente :

« *Comme le préjugé contre l'oral est puissant, il empêche souvent de discerner où sont les zones de grande « complexité » syntaxique* » (Blanche-Benveniste, 2002). Les spécialistes de la langue parlée affirment que la langue parlée répond à ses propres normes, qu'elle est « *organisée autrement* » (que la langue écrite) et, en particulier, que « *la syntaxe est toujours présente dans les réalisations orales, et souvent de façon très complexe* » (Blanche-Benveniste, 2002).

L'étude des réparations fait apparaître comment les hésitations et les retouches procèdent par étoffement des syntagmes nominaux ou verbaux (cf. §4.2) et respectent ainsi la syntaxe locale : par exemple, la reprise de la préposition dans le syntagme nominal permet en général de préserver la suite *préposition-déterminant-nom*. Par ailleurs, C. Blanche-Benveniste affirme que les constructions phrastiques répondent à des critères syntaxiques analogues à ceux de l'écrit et qu'il

est « inutile d'envisager une syntaxe spécifique de l'oral » (Blanche-Benveniste, 2002). Elle ajoute cependant « qu'il existe des réalisations particulières de la syntaxe et des fréquences d'occurrence spécifiques ». Quelques particularités fréquentes sont énumérées par les différents auteurs :

- Le « télescopage syntaxique » de deux constructions syntaxiques existantes :
« *C'est une chose à laquelle on s'habitue très vite à repérer* » (Kerbrat-Orecchioni, 1990)
- La liaison entre lexique et grammaire est plus forte à l'oral ; ainsi l'emploi des pronoms relatifs tels que *dont*, *auquel* est statistiquement très lié à l'usage de certains verbes (« *dont je parle* ») (Blanche-Benveniste, 2002).
- Les sujets des verbes sont bien plus souvent des pronoms que des substantifs nominaux placés devant le verbe (« *il viendra cinq personnes* » plutôt que « *cinq personnes viendront* »).
- Plus ou moins proscrites selon les normes de l'écrit, les répétitions dites lexicales sont très fréquentes dans les énoncés de l'oral spontané (Blanche-Benveniste, 2002) :
« *c'est pour une réservation je voudrais réserver un hôtel...* ».
- L'ordre des mots est un problème très complexe dont les règles sont mal connues. D'après A. Abeillé et D. Godart, les positions relatives entre mots seraient liées à leur *poids lexical* (Abeillé and Godard, 2000). Si l'on excepte les textes poétiques, l'expression orale contient un plus grand nombre de dislocations, appositions, etc. que les productions écrites. Ces phénomènes correspondent souvent à un enchaînement thématique :
« *l'hôtel Caumartin pour une chambre double le prix c'est quoi* ».
- Les changements soudains de parcours syntaxiques ou sémantiques sont fréquents avec, par exemple, les énoncés « parenthétiques » qui peuvent s'insérer dans des positions inattendues (Blanche-Benveniste, 2002) :
« *le bus qui va au Louvre [j'aimerais bien y aller cet après-midi] il se prend où* ».

La majorité des phénomènes observés qui relèvent des deux dernières catégories sont souvent destinés à renforcer l'efficacité de l'interaction. Ainsi, la dislocation à gauche sert à « *promouvoir un référent au statut de topic* » (Pekarek-Doehler, 2001). De même, les organisations en strates successives aident à la compréhension du contenu sémantique de l'énoncé :

« *Le soir du jour qui précède la fermeture hebdomadaire du mercredi / Le jour le soir de la fermeture parce que on ferme le mercredi le mardi soir* ».

Dans ces constructions, le terme sémantiquement inclusif précède le terme inclus, introduit sans préposition (Blanche-Benveniste, 2002) :

« *l'Espagne les hôtels sont chers* ».

Enfin, C. Kerbrat-Orecchioni rapporte une étude de Goodwin selon laquelle les auto-interruptions passagères d'un locuteur coïncident avec une baisse d'attention de son auditeur ; le rôle de l'interruption serait alors de reconquérir cette attention (Kerbrat-Orecchioni, 1990).

Ces observations plaident pour une « *conception interactionniste de la grammaire* », avec un traitement des « *indices grammaticaux comme des instruments que les interlocuteurs utilisent pour se manifester mutuellement leurs interprétations des activités du discours* » (Pekarek-Doehler, 2001).

4.5 Expression orale dans le dialogue Homme/Machine

J. Glass souligne que concevoir et améliorer les systèmes pose le problème de la poule et l'œuf (Glass, 1999) car développer les systèmes demande de disposer de larges corpus de données dans des conditions réelles d'utilisation. Or, ces corpus ne peuvent exister que si le système existe lui-même. Une solution classique est alors de partir d'un corpus obtenu par la méthode du magicien d'Oz (cf. page 24) puis de développer et améliorer itérativement le système avec des données réelles (VOYAGER, JUPITER, etc.).

Il est en effet difficile de savoir comment vont s'exprimer les utilisateurs d'un système avant qu'il ne soit mis en service et la méthode adoptée semble donc raisonnable. Une question se pose néanmoins concernant les améliorations apportées itérativement sur la base de corpus recueillis lors de l'utilisation effective de systèmes. En effet, les modes d'expression des utilisateurs d'un système peuvent varier avec le temps et en particulier, s'adapter aux capacités de compréhension de ce système et à ses déficiences. Par conséquent, on peut craindre qu'ils ne soient pas le reflet de modes d'expression « naturels » et que cette méthode, si elle est utilisée seule, n'ignore certains défauts majeurs.

Cette interrogation montre l'intérêt que peut présenter une étude générale de l'expression orale en situation de dialogue et même si les paragraphes précédents concernent des dialogues homme-homme, on peut en tirer sans trop de risques quelques conclusions importantes :

- Les marques du travail de formulation, lorsqu'elles concernent les syntagmes, n'ont pas de raison d'être absentes des énoncés des utilisateurs d'un système de DOHM. Pour l'analyse automatique, on peut retenir qu'elles respectent l'ordre des mots et la structure syntaxique interne du syntagme, et que les reprises concernent généralement les syntagmes complets.
- Beaucoup des phénomènes observés dans les constructions phrastiques, par exemple les constructions syntaxiques télescopées et les dislocations, sont des marques d'interaction, et, comme tels, risquent d'être moins fréquents en DOHM. Concernant les mouvements des constituants, l'un des enjeux essentiels pour une analyse automatique est de savoir à quel type de variabilité ils appartiennent car cette propriété conditionne la possibilité d'utilisation de certains formalismes (en particulier, celui des grammaires de liens (Sleator and Temperley, 1991)). La variabilité dite *faible* correspond à un mouvement des constituants qui respecte la *projectivité* : les liens entre les constituants ne se croisent pas. Au contraire, la variabilité dite *forte* entraîne un croisement des liens entre les constituants (Holan et al., 1998; Holan et al., 2000).

J-Y. Antoine et J. Goulian, à l'issue d'une étude de deux corpus Homme/Homme, concluent à une très faible présence de la variabilité forte -présente mais de façon marginale - et à la nécessité de prendre en compte la variabilité faible, de par sa fréquence élevée (Antoine and Goulian, 2001).

4.6 Conclusion

Le matériau sur lequel doit travailler la compréhension pose des problèmes de différentes natures qui sont liés soit à la reconnaissance vocale, soit aux spécificités de l'expression orale spontanée.

Les erreurs liées à la reconnaissance de la parole perturbent le message envoyé par le locuteur. Dans la retranscription orthographique, les phénomènes correspondants peuvent être qualifiés de « bruit » car ils ne répondent à aucune règle syntaxique ou sémantique. Seule la robustesse de l'analyse peut permettre d'y faire face.

Les spécificités de l'expression orale spontanée posent des problèmes pour une application directe de certaines techniques utilisées dans l'analyse de l'écrit. Cependant, les études faites par les linguistes montrent qu'elles semblent répondre à certaines règles. Les analyses des énoncés peuvent donc mettre en œuvre des traitements qui permettent d'exploiter ces observations.

Deuxième partie

Logus

Chapitre 5

Principes généraux et première mise en œuvre

LOGUS est un système de compréhension des énoncés oraux dans le contexte d'un dialogue Homme-Machine finalisé. L'application test choisie est celle de l'interrogation d'une base de données dans le domaine du renseignement touristique. Bien que délimité, ce domaine est plus vaste et plus riche que la plupart de ceux des systèmes classiques et il permet une grande variété de requêtes.

Le cadre applicatif consiste à extraire le sens d'un énoncé à partir d'une liste de mots préalablement reconnus. Ce choix répond à une volonté délibérée de construire un module de compréhension indépendant du système de reconnaissance. S'il exclut d'emblée la possibilité d'inférences entre les deux modules pour limiter le nombre des erreurs de la reconnaissance ou leurs effets, il a pour avantages évidents de permettre une plus grande modularité et une meilleure généralité. Par ailleurs, il est également justifié par les progrès rapides de la reconnaissance vocale et la volonté de faire une compréhension un peu prospective par rapport aux systèmes actuellement opérationnels.

Les deux premières parties de ce chapitre présentent les objectifs qui ont présidé à l'élaboration du système et les principes initiaux de l'approche utilisée. La troisième partie décrit une première mise en œuvre de ces principes et leur partielle remise en cause.

5.1 Les objectifs

L'objectif essentiel est de proposer une représentation du sens des énoncés et une méthode pour construire cette représentation. Cette méthode doit être applicable aux énoncés de l'oral spontané concernant un domaine relativement circonscrit où une connaissance sémantique des objets concernés peut être définie. A priori, l'approche serait également adaptable à la compréhension d'énoncés écrits dans une langue non contrainte et dans un contexte similaire.

Dans l'analyse du sens, LOGUS prétend concilier deux objectifs a priori antagonistes : la finesse et la robustesse.

Finesse de l'analyse : l'analyse se veut « fine » parce qu'elle prétend rendre compte d'une manière aussi précise que possible du sens de l'énoncé. Elle se trouve donc confrontée à une double

complexité :

- Pour pouvoir traduire aussi fidèlement que possible les intentions du locuteur, elle doit pouvoir analyser des indices qui permettent de détecter une large palette d’actes de dialogue.
- Elle doit également être capable de reconnaître et de modéliser les objets liés au domaine ainsi que leurs propriétés. Le renseignement touristique permet d’envisager une certaine complexité de ces objets (« *le tarif des chambres doubles ou simples au Caumartin et au Crillon* »).

Cette exigence sous-entend la prise en compte du plus grand nombre possible des éléments de l’énoncé. Conserver un maximum de souplesse implique également l’abandon de structures sémantiques préfabriquées (schémas avec structures de traits) et donc nécessite que soit définie une langue cible qui permette une représentation sémantique à la fois souple et précise.

Robustesse : en même temps, l’analyse se veut « robuste » et donc capable d’analyser le matériau qui lui est soumis, décrit dans le chapitre 4 de la partie I. Elle doit pouvoir résister :

- aux phénomènes relatifs au « travail de formulation » : hésitations, reprises, auto-corrrections, etc.
- aux phénomènes relatifs à l’interaction : « phatiques » qui émaillent les énoncés oraux : « *quoi* », « *bien* », « *bon* », etc.
- à la souplesse de l’organisation syntaxique ou sémantique des énoncés oraux : organisation en strates successives, mouvements des constituants (dislocations, structures clivées ou pseudo-clivées etc.), changements de parcours syntaxiques et sémantiques (faux-départs, incises, etc.),
- et, bien sûr, à un nombre non négligeable d’erreurs de reconnaissance.

Il s’agit donc de trouver un juste équilibre entre l’obligation de pouvoir ne pas tenir compte de certains éléments (robustesse oblige) et la volonté de traduire aussi précisément que possible ce que le locuteur veut dire (finesse oblige). En particulier, la langue cible se doit d’être suffisamment expressive dans le domaine concerné. En même temps, l’analyse ne doit pas pour autant permettre des constructions sémantiquement incohérentes.

5.2 Représentation sémantique des énoncés

5.2.1 Introduction

La représentation sémantique d’un énoncé est un élément déterminant du système ; elle définit la façon dont sont conçus le sens de cet énoncé et l’usage que l’on peut en faire. Selon Minker,

« *le développement d’un analyseur sémantique n’est [...] pas limité à l’extraction du sens mais doit également prendre en compte la manière dont la représentation sémantique, telle que sortie de l’analyseur, est utilisée ultérieurement dans le système* » (Minker, 1999).

La représentation sémantique est l’image donnée par le système du sens de l’énoncé. Elle doit donc être conçue en fonction des besoins pressentis du système de dialogue. Or, le but de ce dialogue est d’accomplir une ou plusieurs tâches, selon les désirs de l’utilisateur : la représentation sémantique doit donc pouvoir rendre compte de tout ce qui peut aider à mener à bien ces tâches.

Ainsi, les travaux concernant la compréhension dans ce type de systèmes font parfois état du « *sens utile* » de l'énoncé pour justifier la possibilité de négliger certains de ses éléments (Pérennou, 1996).

La délimitation précise entre ce qui peut être *utile* et ce qui ne l'est pas peut prêter à discussion et dépend directement du domaine et des tâches que couvre le système. Par exemple, dans l'énoncé « *j'ai la grippe où est la pharmacie la plus proche* », on peut penser que la première information n'a rien d'« *utile* » dans un système de renseignements touristiques. Cependant, les choses ne sont pas toujours aussi claires. Dans l'énoncé « *je voudrais aller au Louvre où peut-on acheter un ticket de métro* », la première information n'est pas directement liée à la requête ; elle peut néanmoins être utile pour la poursuite du dialogue, si par exemple l'utilisateur veut des renseignements concernant la ligne de métro à prendre ou la visite au musée qu'il se propose de faire. Il n'est donc pas évident de déterminer a priori ce qui est ou sera utile au bon fonctionnement du dialogue, si l'on veut laisser ouvertes un maximum de possibilités pour son déroulement ultérieur.

De toute façon, la compréhension du système est limitée par la « connaissance » qu'il a de son domaine d'application, restreint par construction. Le principe adopté dans LOGUS est simple : il consiste à considérer que la connaissance du domaine et la gestion du dialogue déterminent ce qui doit être pris en considération. Le sens « utile » inclut donc tout ce qui appartient au monde connu du système¹, l'inutile étant ce qui est inconnu. Ainsi, la représentation sémantique proposée doit pouvoir rendre compte

- de tout ce qui peut servir à décrire les objets du domaine ou leurs propriétés,
- de tout ce qui peut permettre de reconnaître les intentions du locuteur.

Les informations qui ne sont pas immédiatement utiles pour satisfaire les désirs de l'utilisateur peuvent être mises en contexte pour une poursuite éventuelle du dialogue.

Le choix de la forme de la représentation sémantique fait dans LOGUS est celui d'une formule logique à la Montague, avec un formalisme simplifié (Montague, 1974). La formule est le résultat de la composition d'un éventuel acte de langage (ou du moins des indices de cet acte) avec les objets et les propriétés du domaine figurant dans l'énoncé. Cette représentation se rapproche de celle proposée par D. Vanderveken dans sa logique illocutoire (cf. page 41). Le *contenu propositionnel* correspond aux objets du domaine ; on peut lui attribuer une valeur de vérité correspondant à leur existence. La *force illocutoire* est représentée par un prédicat qui traduit les indices permettant d'inférer l'acte de langage. La formule est également conçue pour être aisément transcribable en un graphe conceptuel à la Sowa (Sowa, 2000).

Par exemple, les deux énoncés suivants correspondent à la même forme propositionnelle mais à des forces illocutoires différentes :

- (1) « *j'ai réservé une chambre double à l'hôtel Caumartin* »
- (2) « *est-ce que c'est possible de réserver une chambre double à l'hôtel Caumartin* »

La forme propositionnelle correspond à une « *réservation d'une chambre double à l'hôtel Caumartin* ». Le premier énoncé exprime une information d'existence effective alors que le second correspond à une demande d'information.

Pour parvenir à une telle représentation, le langage cible doit donc permettre de construire une structure qui permet de modéliser l'objet complexe concerné d'une part (désigné sous le terme

1. Les 1400 énoncés du corpus PARISCORP, élaborés dans le cadre de l'Action de Recherche Concertée B2 de l'AUF (Agence Universitaire de la Francophonie), ont permis d'avoir une première approximation du domaine et de la tâche (Bonneau-Maynard and Devillers, 1998; Rosset et al., 1997).

générique de *chaîne d'objets*) et de rendre compte des indices permettant de reconnaître l'acte de langage d'autre part.

5.2.2 Représentation des objets du domaine

Introduction

La structure qui correspond à « *la réservation d'une chambre double à l'hôtel Caumartin* » est formée en établissant un lien de dépendances (dite *relation d'enchaînement*) entre les trois objets élémentaires *réservation*, *chambre double* et *hôtel Caumartin*. Chacun de ces objets, dit *simple*, est caractérisé par sa nature, qui est représentée par ce que nous appellerons son *étiquette* (*réservation*, *chambre*, *hôtel*) et la liste de ses propriétés : *double* est la détermination d'une propriété de la chambre liée à sa *taille*, *Caumartin* est le nom de l'hôtel qui permet de déterminer son *identité*. Les termes de *taille* et d'*identité* renvoient à la nature de la propriété (*étiquette* de la propriété) ; *double* et *Caumartin* permettent de la déterminer (*détermination* de la propriété).

Pour compléter la manipulation des objets complexes, d'autres relations de dépendances doivent être définies, en particulier pour exprimer les coordinations. Ainsi « *les tarifs du Louvre et du Grévin* » conduit à la coordination de deux chaînes d'objets : la première correspond à l'objet « *les tarifs du Louvre* », la seconde à l'objet « *les tarifs du Grévin* ».

La notion d'*objet* recouvre des entités concrètes du domaine (tels que les hôtels, restaurants, musées, etc.), certains de leurs éléments (chambres, repas, menus, etc.), des entités abstraites ou concrètes qui leur sont liées (visites, réservations, locations, paiements, etc.) ou plus générales mais considérées néanmoins comme signifiantes ou *utiles* au sens qui a été défini (perte, oubli, etc.).

Les propriétés correspondent à des déterminations des objets, qui ne peuvent exister qu'en tant que telles. Décider si un concept doit appartenir au monde des objets ou à celui des propriétés peut ne pas aller de soi : par exemple le concept *étoiles*², initialement placé dans le monde des objets est actuellement considéré dans LOGUS comme une simple propriété : il ne possède aucun élément et il n'intervient que s'il est lié à un entier pour caractériser la catégorie d'un hôtel (ou d'un camping). L'intérêt de la distinction entre objets et propriétés tient essentiellement en ce qu'elle facilite la définition de la connaissance sémantique, celle des règles syntaxico-sémantiques et la manipulation des chaînes d'objets au cours de l'analyse. Elle répond donc à des motivations technologiques plutôt que théoriques. On peut noter d'ailleurs qu'elle disparaît lorsque la représentation sémantique finale est donnée sous la forme d'un graphe conceptuel (cf. §5.2.5).

La relation d'enchaînement modélise un lien entre les objets qui correspond à une appartenance ou à une inclusion. Elle ne suffit pas à exprimer l'ensemble des dépendances possibles entre objets. Par exemple, dans l'expression « *un hôtel pas loin du Louvre* », l'objet « *Louvre* » est utilisé pour préciser le lieu de l'objet « *un hôtel* ». Dans LOGUS, cette dépendance est modélisée en faisant de « *pas loin du Louvre* » une propriété de *proximité* (*étiquette* de la propriété) dont la *détermination* est l'objet *musée du Louvre*. Ainsi, un *objet* peut être utilisé comme la *détermination* d'une propriété. Cette possibilité conduit à des définitions *mutuellement récursives* des objets et de leurs propriétés dont le formalisme est exposé dans le paragraphe suivant.

2. Le concept *étoiles* doit être distingué du lexème « *étoiles* » contenu dans le lexique ; ce lexème peut en effet s'associer à un déterminant et à un entier pour former des expressions telles que « *un deux étoiles* » qui désignent un objet d'*étiquette* *hôtel*.

Définition des données

Des ensembles de *constants* permettent de symboliser les objets et leurs propriétés. Des *relations* servent à exprimer leurs dépendances.

1. $\mathcal{E}t_Obj$ est l'ensemble des *étiquettes des objets*. Cet ensemble dépend directement du domaine de l'application ; il comprend dans la version actuelle de LOGUS une centaine d'éléments :

$$\mathcal{E}t_Obj = \{ \text{hotel, restaurant, musee, ...}, \\ \text{chambre, repas, visite, ...}, \text{location, tarif, ...} \}$$

2. $\mathcal{E}t_Prop$ est l'ensemble des *étiquettes des propriétés*. Il contient actuellement une trentaine d'éléments qui sont relativement indépendants du domaine de l'application :

$$\mathcal{E}t_Prop = \{ \text{prix, lieu, horaire, quantite, identite, taille, nbetoiles, ...} \}$$

3. Sem_Prop est l'ensemble des *déterminations des propriétés*. Une grande partie d'entre elles dépendent très fortement du domaine de l'application. Certaines d'entre elles s'obtiennent avec des constructeurs tels que

- *entier* qui prend en entrée un entier naturel,
- *nom* qui prend en entrée une chaîne de caractères,
- *euros* qui prend en entrée une détermination de propriété (qui représente un nombre)

$$Sem_Prop = \{ \text{avec_douche, cher, eleve, double, ...}, \\ (\text{entier } _), (\text{nom } _), (\text{euros } _) \dots \}$$

Des *modifieurs* peuvent être appliqués à ces déterminations : ils constituent l'ensemble Mod_det_prop .

$$Mod_det_prop = \{ \text{infegal, pas, peu, environ, etc.} \}$$

4. La *relation d'enchaînement entre objets* est symbolisée par *de*.
5. Rel_obj_Prop est l'ensemble des *relations d'objet à propriété*. Elles jouent le rôle d'étiquettes de propriétés dont la détermination est un objet.

$$Rel_obj_Prop = \{ \text{proche_de, vers, venant_de, avec, sans, etc.} \}$$

6. $Coord_obj$ est l'ensemble des *coordinations entre objets*.

$$Coord_obj = \{ \text{et, ou, entre, et_pas, etc.} \}$$

Définition des propriétés

Définition : une *propriété* est

1. Soit un couple $(Et_prop, Sem_prop) \in \mathcal{E}t_Prop \times Sem_Prop$.
Et_prop est donc une étiquette de propriété et *Sem_prop* est une détermination de cette propriété.

Exemples :

- « au moins deux » $(quantite, (supegal (entier 2)))$
- « pas cher » $(prix, (pas cher))$
- « moins de cinquante euros » $(prix, (infegal (entier 50)))$

2. Soit un couple (Rel_obj_prop, Obj) où *Rel_obj_prop* $\in \mathcal{R}el_obj_Prop$ (une relation d'objet à propriété) et *Obj* est un objet ou une chaîne d'objets.

Exemple :

- « près du Louvre » $(proche_de, (musee [(identite (nom "Louvre"))]))$

Cette deuxième définition utilise donc le concept d'objets qui est défini ci-après.

Objets et des chaînes d'objets

Définitions :

1. Un *objet simple* est un couple (Et_o, L_prop) où *Et_o* est un élément de $\mathcal{E}t_Obj$ (une étiquette d'objet) et où *L_prop* est une liste de propriétés.

Exemple : « un hôtel deux étoiles pas cher »

$$(hotel, [(prix, (pas cher)), (nbetoiles, (quantite, (entier 2))]))$$

2. Une *chaîne d'objets simple* est un *objet simple* ou un triplet $\langle de, O_1, O_2 \rangle$ où O_1 est un *objet simple* et où O_2 est une *chaîne d'objets simple*.
3. Une *chaîne d'objets composée* est un triplet $\langle Coord, CO_1, CO_2 \rangle$ où *Coord* est un élément de *Coord_obj* (une coordination entre objets), CO_1 et CO_2 des *chaînes d'objets simples* ou bien un triplet $\langle de, O_1, O_2 \rangle$ où O_1 et O_2 sont des chaînes d'objets dont l'une au moins est *composée*.

Exemple 1 : « les tarifs du Louvre et du Grévin »

$$\langle de, (tarif, []), \langle et, (musee, [(identite (nom "Louvre"))]), (musee, [(identite (nom "Grevin"))]) \rangle \rangle$$

Exemple 2 : « les horaires du musée du Louvre et les tarifs du Grévin »

$$\langle et, \langle de, (horaire, [(date, (nom_jour dimanche))]), (musee, [(identite (nom "Louvre"))]), \langle de, (tarif, []), (musee, [(identite (nom "Grevin"))]) \rangle \rangle \rangle$$

Opérations sur les chaînes

Le formalisme développé dans ce paragraphe est utilisé dans l'analyse des énoncés pour manipuler les chaînes d'objets et pour construire la représentation sémantique finale. Il introduit les notions suivantes :

1. Forme développée d'un chaîne

Les *formes développées* d'une chaîne sont utilisées pour la compréhension des requêtes multiples ; par exemple, la chaîne d'objets suivante est la forme développée de l'exemple 1 précédent :

« *les tarifs du Louvre et du Grévin* » :

$$\langle \text{et}, \langle \text{de}, (\text{tarif}, []) \text{ (musee, [(identite (nom "Louvre")])]), \\ \langle \text{de}, (\text{tarif}, []) \text{ (musee, [(identite (nom "Grevin")])]) \rangle \rangle$$

Cette forme permet de faire apparaître deux chaînes d'objets complètes et de faire porter les actes de langage sur l'une et l'autre de ces chaînes.

Définition :

Une chaîne d'objets est *sous forme développée* si et seulement si elle vérifie l'une des conditions suivantes :

- (a) elle est simple,
- (b) elle est sous la forme $\langle \text{Coord}, CO_1, CO_2 \rangle$ où $\text{Coord} \in \text{Coord_obj}$, où CO_1 est une chaîne simple et CO_2 est une *chaîne développée*.

Théorème :

Toute chaîne d'objets peut être mise sous forme développée et cette forme est unique, à l'ordre des chaînes simples obtenues près.

Ce théorème est évident par induction.

2. Objets minimaux

La notion d'*objets minimaux* est définie pour désigner les objets situés à *gauche* dans les chaînes : ils correspondent aux objets sur lesquels porte l'acte de langage. Par exemple dans l'énoncé :

« *pour une chambre double vous pouvez me donner les tarifs s'il vous plaît* », la requête porte sur *les tarifs* qui correspond à l'*objet minimal* de la chaîne :

$$\langle \text{de}, (\text{tarif}, []) \text{ (chambre, [(taille, double)])} \rangle$$

Définition :

- L'*objet minimal* d'un objet simple est lui-même.
- L'*objet minimal* de la chaîne simple $\langle \text{de}, O_1, O_2 \rangle$ est O_1 .
- L'ensemble des objets minimaux de la chaîne *développée* $\langle \text{Coord}, CO_1, CO_2 \rangle$ est $\{O_1\} \cup MO_2$ avec O_1 élément minimal de CO_1 et MO_2 ensemble des éléments minimaux de CO_2 .

3. Étiquette d'une chaîne

L'étiquette d'une chaîne correspond à l'étiquette de l'ensemble de ses objets minimaux, lorsque cette étiquette est unique ; par exemple, la chaîne d'objets qui correspond à « l'adresse et les tarifs du Caumartin »

$$\begin{aligned} &< et, < de, (adresse, []) (hotel, [(identite, (nom "Caumartin"))]) > \\ &\quad < de, (tarif, []) (hotel, [(identite, (nom "Caumartin"))]) > \\ &> \end{aligned}$$

admet deux objets minimaux (*adresse*, []) et (*tarif*, []) d'étiquettes distinctes. Elle n'a donc pas d'étiquette.

Définition :

Soit CO une chaîne d'objet et MO l'ensemble de ses objets minimaux. Si $MO = \{O\}$ (singleton), alors l'étiquette de la chaîne CO est celle de O ; si MO contient plusieurs éléments minimaux, alors, CO possède une étiquette Et si et seulement si chacun de ces objets minimaux admet Et pour étiquette.

4. Segment droit

Les notions de *segment droit* et de *factorisation* sont définies pour le traitement des coordinations (cf. page 106) et pour l'interprétation contextuelle (cf. page 112).

Les segments droits d'une chaîne correspondent à ses sous-chaînes droites : par exemple, la chaîne qui correspond à l'expression :

« les tarifs d'une chambre double à l'hôtel Caumartin » possède 3 segments droits qui sont elle-même et les chaînes correspondant à : « une chambre double à l'hôtel Caumartin » et « l'hôtel Caumartin ».

Définition : Le seul *segment droit* d'un objet simple est l'objet lui-même.

Un *segment droit* d'un objet O de la forme $< de, O_1, O_2 >$ est O lui-même ou un segment droit de O_2 .

5. Factorisation

La factorisation d'une chaîne composée consiste à compléter à droite l'une des chaînes coordonnées. Par exemple :

$$\begin{aligned} &(et (adresse, []) \\ &\quad < de, (tarif, []), (hotel, [(identite, (nom "Caumartin"))]) >) \end{aligned}$$

devient, par factorisation :

$$\begin{aligned} &(et < de, (adresse, []), (hotel, [(identite, (nom "Caumartin"))]) > \\ &\quad < de, (tarif, []), (hotel, [(identite, (nom "Caumartin"))]) >) \end{aligned}$$

Définition : Étant donnée une chaîne d'objets O de la forme $< Coord, O_1, O_2 >$ où $Coord \in Coord_obj$ ($Coord$ est une coordination entre objets) et où O_2 est un objet simple, *factoriser O à droite* consiste à remplacer O_2 par $< de, O_2, SO_1 >$ où SO_1 est un segment droit de O_1 différent de O_1 .

Ces *factorisations* sont faites sous le contrôle de la connaissance sémantique.

5.2.3 Reconnaissance et représentation des actes de langage

La détermination des actes de langage pose deux problèmes qui conditionnent le choix de leur représentation :

- Tout d’abord, il convient de préciser la nature des actes³ auxquels on doit s’attendre dans le dialogue envisagé, compte tenu du domaine d’application concerné.
- Ensuite, et même si le but final est une compréhension complète et en contexte des intentions du locuteur, on peut se demander également à quel niveau de la compréhension il est possible de déterminer leur nature exacte.

Étude de la nature des énoncés

Les énoncés qui engagent le dialogue sont a priori des requêtes, donc des actes *directifs* au sens de Searle (cf. page 40). Ces requêtes peuvent être de deux natures différentes : la plupart du temps, elles correspondent à une demande d’informations qui concerne l’état de la base de données. Cependant, elles peuvent également conduire à une modification de l’état de cette base ; dans l’application choisie, ces dernières requêtes correspondent à des demandes de réservation ou de location.

– Demandes d’informations :

Une classification classique des demandes d’information consiste à tenir compte du type des réponses attendues.

- Dans les *requêtes fermées*, la réponse attendue est du type *oui-non*, comme par exemple dans l’énoncé :

« *Est-ce que l’hôtel Caumartin est accessible ?* »

- Dans les *requêtes ouvertes*, la réponse attendue est une liste d’objets ou de propriétés répondant à certaines conditions :

« *Quels sont les tarifs de l’hôtel Caumartin ?* » .

La réponse peut être aussi la description d’une propriété donnée :

« *Combien d’étoiles il a l’hôtel Caumartin ?* » .

– Demandes de réservation ou location :

Dans l’application, les demandes dont le succès correspond à une modification de l’état de la base de données peuvent être des réservations ou des locations de véhicules, de chambres d’hôtel, etc.

« *Réservez-moi une chambre double pour mardi prochain* » .

– Autres actes de dialogue :

- Le locuteur peut exprimer son accord, apporter des corrections ou opposer un refus à ce qu’il pense que le système a compris de sa requête ou aux propositions qui lui sont faites :

« *Oui d’accord pour un deux étoiles* »,

« *Plutôt deux étoiles s’il vous plaît* »,

« *Non ce n’est pas un deux étoiles que je veux* » .

3. Des considérations générales ont déjà été exposées page 47.

- Il peut exprimer des faits sur le monde (ce qui correspond à des actes *assertifs* au sens de Searle) qui contiennent des informations utiles au contexte du dialogue :
« *J'ai réservé à l'hôtel Caumartin* » .
- Il peut enfin exprimer des intentions :
« *Je voudrais visiter le Louvre* » .
- Les actes *expressifs* peuvent être présents sous la forme de remerciements ou d'expressions de mécontentements. Leur intérêt éventuel est de pouvoir signifier un bon ou un mauvais déroulement du dialogue.
- Il convient enfin d'envisager la possibilité d'exprimer successivement plusieurs actes de langage successifs dans le même énoncé.
Par exemple, *information* suivie de *requête*,
« *Je voudrais visiter le Louvre comment je peux y aller d'ici* »
refus suivi de *confirmation* ou *requête*,
« *Le Caumartin me convient bien est-ce qu'il y a un restaurant pas loin* »,
etc.

Malgré la quasi-absence de tels énoncés dans les corpus étudiés , il nous semble qu'il s'agit là d'un mode d'expression « naturel » relativement fréquent.

Détermination de la nature des énoncés

L'analyse de l'énoncé hors contexte ne permet pas à elle seule de distinguer les différents types d'énoncés tels qu'ils viennent d'être définis, et ce, pour plusieurs raisons :

- Les requêtes sont très souvent émises sous forme *indirecte* : « *pouvez-vous me donner l'adresse du Caumartin* » n'attend pas une réponse en *oui-non* comme semblerait l'indiquer sa forme linguistique.
- Dans le déroulement du dialogue, beaucoup d'énoncés sont elliptiques ou syntaxiquement incomplets : un énoncé tel que « *deux si possible* » fait référence à des objets du contexte et ne peut s'interpréter en dehors de ce contexte.
- Même dans le cas où l'énoncé est syntaxiquement complet, il n'est pas toujours possible de déterminer s'il s'agit d'une forme affirmative (par exemple, réponse à une question posée) ou d'une forme interrogative *totale* (cf. page 55) : ainsi l'énoncé « *les couples ont droit à une réduction* » peut être compris soit comme une interrogation soit comme une affirmation à faire valoir un avantage. Si l'on ne dispose pas d'indications prosodiques, seul le contexte de dialogue peut éventuellement permettre de lever cette ambiguïté.
- Même dans le cas d'une requête directe, il n'est pas évident de déterminer s'il s'agit d'une requête ouverte ou d'une requête fermée. Une réponse de type *oui-non* peut satisfaire l'utilisateur à la question :
(1) « *est-ce qu'il y a des réductions pour les étudiants au Gaumont-Opéra* »
Il est par contre douteux qu'une telle réponse le satisfasse s'il pose la question :
(2) « *est-ce qu'il y a des films en VO au Gaumont-Opéra* »
On peut en effet remarquer que si la réponse était affirmative, un opérateur humain la ferait immédiatement suivre de la liste des films concernés, sans attendre la question suivante : « *lesquels ?* » que ne manquerait pas de poser l'interlocuteur.

On peut tirer quelques conclusions des exemples précédents concernant le niveau auquel peut se faire la compréhension précise des intentions du locuteur.

- La nature exacte des requêtes émise sous forme indirecte peut en général se déduire de l'objet sur lequel elle porte (objet *minimal* de la chaîne). Par exemple une information sur une adresse est a priori une requête ouverte (« *pouvez-vous me donner l'adresse...* »).
- Les énoncés elliptiques et les formes interrogatives totales exigent une interprétation contextuelle, qu'elle soit opérée par le module de compréhension ou par le module de dialogue.
- Concernant la distinction entre les requêtes ouvertes et les requêtes fermées, le choix semble dépendre des réponses possibles, et donc de la base de données. Ainsi la question (1) précédente peut être classée dans les requêtes fermées si les étudiants ont droit à un seul type de réduction puisqu'alors, une réponse de type *oui_non* peut suffire. Si par contre, la réduction faite aux étudiants est variable (si elle dépend des jours de la semaine par exemple), la meilleure réponse à donner est la liste des réductions possibles et l'on peut alors considérer la requête comme une requête ouverte. En définitive, seul le module de dialogue, selon les réponses présentes dans la base de données peut donc prendre la décision de la meilleure forme de la réponse à apporter à la question posée. Ces remarques font que l'on peut se demander si cette distinction entre requêtes ouvertes ou fermées est réellement pertinente dans le contexte de l'application envisagée.

Il est donc clair que de toute manière, la nature précise de l'acte ne peut se déduire en général de la seule forme linguistique ; sa détermination dépend de la nature des objets concernés et très souvent du contexte. Il convient donc de prévoir une représentation des constituants qui permette de modéliser au plus près les indices linguistiques des actes de langage pour leur détermination ultérieure.

Représentation des (indices concernant les) actes de langage

L'expression « *est-ce qu'il y a* » des exemples (1) et (2) précédents indique une *interrogation* sur une *existence*. Dans LOGUS, la représentation de la *force illocutoire* utilise deux paramètres qui marquent simplement l'existence d'une forme *interrogative* (appelée *forme expressive*) en *il_y_a* (le *contenu expressif*).

Formellement, la représentation des indices est fondée sur les deux ensembles suivants :

- \mathcal{F}_{acte} est l'ensemble des *formes expressives*. Ses éléments indiquent ce que la forme syntaxique de l'énoncé peut permettre de savoir de la nature des phrases analysées.

$$\mathcal{F}_{acte} = \{interrogation, requete, information, expression, modalite, \dots\}$$

- $Cont_{acte}$ est l'ensemble des *contenus expressifs*.

$$Cont_{acte} = \{ubi, quando, quel, \dots, il_y_a, \dots, possibilite, refuser, remercier, \dots\}$$

La nature de l'énoncé, ou du moins ce que l'analyse permet d'en reconnaître, est alors représenté par un couple $\langle Role_{diag}, Cont_{acte} \rangle$ élément de $\mathcal{F}_{acte} \times Cont_{acte}$.

dans la figure 5.2.

Les prédicats qui représentent la nature des énoncés sont également du type ($trad_sem \rightarrow trad_sem$). Les éléments de $Cont_acte$ sont du type $trad_sem$ alors que ceux de \mathcal{F}_acte sont du type $trad_sem \rightarrow (trad_sem \rightarrow trad_sem)$.

Les éléments se composent entre eux pour obtenir une formule sémantique de type $trad_sem$.

Exemple : « *quels sont les hôtels deux étoiles pas trop chers* »

((*interrogation quel*) (*hotel [(prix (pas cher)), (nbetoiles (entier 2))]*))

5.2.5 Graphes conceptuels

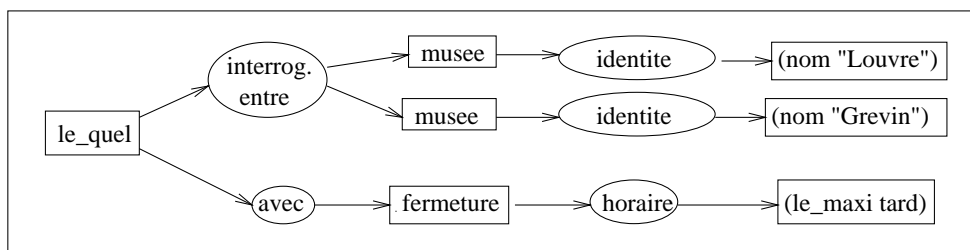
La représentation sémantique de l'énoncé peut également être envisagée comme un graphe conceptuel à la Sowa (Sowa, 2000). Les formules logiques construites sont convertibles en graphes conceptuels de la façon suivante :

- Les *étiquettes des objets*, les *déterminations des propriétés* et les *déterminations des énonciations* constituent les concepts du graphe.
- Les *étiquettes des propriétés*, les *relations d'enchaînements entre objets*, les *relations d'objets à propriétés*, les *coordinations entre objets* et les *rôles sémantiques énonciatifs* sont des relations conceptuelles.

La figure 5.1 donne l'exemple de la représentation sémantique d'un énoncé sous la forme d'une formule logique et sous la forme d'un graphe conceptuel. Selon les conventions habituelles, les concepts sont contenus dans des rectangles alors que les relations conceptuelles sont figurées dans des (pseudo-)ellipses (Sowa, 1984). D'après les définitions données, la chaîne d'objets est une chaîne d'objets simple dont l'objet minimal a pour étiquette *liste*. Sémantiquement, l'objet minimal correspond à celui sur lequel porte la requête.

Le graphe conceptuel de

« *Entre le Louvre et le Grévin lequel ferme le plus tard* » (l'exemple de la page 73) correspond à :



5.2.6 Pouvoir d'expression

Il semble que langage cible choisi pour traduire le sens des énoncés offre des représentations assez simples pour qu'elles puissent être facilement exploitables par le module de dialogue. Le problème qui se pose est de savoir si son pouvoir d'expression est suffisant pour représenter le *sens utile* des énoncés proposés (cf. page 63).

Le *sens utile* des 1400 énoncés du corpus PARISCORP (cf. page 63) et ceux de l'évaluation par défi (cf. page 23), à l'exception de moins d'une dizaine d'entre eux, peuvent être représentés sous

<i>Formule</i>	::=	$(\mathcal{A}cte\ \mathcal{C}haîne_Obj) (\mathcal{A}cte\ \mathcal{P}ropriete) $ $\mathcal{A}cte \mathcal{C}haîne_Obj \mathcal{P}ropriete)$
<i>Acte</i>	::=	$(\mathcal{F}_acte\ \mathcal{C}ont_acte)$
<i>\mathcal{F}_acte</i>	::=	<i>interrogation requete modalite information </i> <i>reprise excuse annexe neglige demande ...</i>
<i>Cont_acte</i>	::=	<i>quel quoi quomodo ubi quando oui_non y_a_t_il </i> <i>refus intention comparaison recherche choix vouloir ...</i>
<i>Chaîne_Obj</i>	::=	$\mathcal{O}bjet_simple (de\ \mathcal{O}bjet_simple\ \mathcal{C}haîne_Obj) $ $(\mathcal{C}oord_obj\ \mathcal{C}haîne_Obj\ \mathcal{C}haîne_Obj)$
<i>Objet_simple</i>	::=	$(\mathcal{E}t_Obj\ (liste\ de\ \mathcal{P}ropriete))$
<i>$\mathcal{E}t_Obj$</i>	::=	<i>hotel restaurant cafe cinema musee monument ...</i> <i>chambre menu carte_restaurant plat_du_jour film ...</i> <i>location ouverture fermeture adresse tarif ...</i> <i>achat liste argent_liquide change_argent ...</i> <i>jour nuit vol perte oubli ...</i> <i>manger dormir prendre attendre ...</i>
<i>Coord_obj</i>	::=	<i>et ou et_pas entre ...</i>
<i>Propriete</i>	::=	$(\mathcal{E}t_prop\ \mathcal{S}em_prop) (\mathcal{R}el_obj_prop\ \mathcal{C}haîne_Obj)$
<i>$\mathcal{E}t_prop$</i>	::=	<i>lieu direction niveau taille nbetoiles qualite quantite </i> <i>date horaire temps_duree identite numero quantificateur ...</i>
<i>Sem_prop</i>	::=	$\mathcal{E}lem_sem_prop (\mathcal{M}od_det_prop\ \mathcal{S}em_prop) $ $(\mathcal{C}ompar\ \mathcal{S}em_prop\ \mathcal{C}haîne_Obj)$
<i>Elem_sem_prop</i>	::=	<i>bas moyen leve double simple avec_douche </i> <i>maintenant disponible accessible ouvert ferme tous_toutes </i> $(entier\ integer) (nom\ string) (jours\ snb\ \mathcal{S}em_prop) ...$
<i>Mod_det_prop</i>	::=	<i>pas environ supegal in_fegal plus moins </i> <i>avant apres tres peu lemaxi lemini ...</i>
<i>Compar</i>	::=	<i>plus_que moins_que meme_que autre_que</i>
<i>Rel_obj_prop</i>	::=	<i>avec sans proche_de vers venant_de ...</i>

FIG. 5.2 – Grammaire des termes de la représentation sémantique

la forme qui vient d'être présentée : un acte de langage appliqué à une chaîne d'objets simple ou complexe. Les énoncés elliptiques dont le sens ne peut être interprété qu'en fonction du contexte de dialogue et les énoncés dont une partie importante a été mal identifiée par la reconnaissance vocale peuvent conduire à des formes incomplètes. L'acte de langage peut ne pas être présent : « *les tarifs d'un hotel deux étoiles* » et/ou la chaîne d'objets peut se réduire par exemple à une propriété : « *pas trop loin du Louvre* ».

Dans ces deux corpus, les seuls énoncés qui n'entrent pas dans ce cadre appartiennent à une catégorie d'énoncés que l'on peut qualifier de « *comparatifs* ». Par exemple, l'énoncé suivant est extrait du corpus de l'évaluation « par défi » (cf. pages 23 et 115) :

« *Entre le Louvre et le Grévin lequel ferme le plus tard* »

Dans cet exemple, la requête porte sur la propriété « *ferme le plus tard* » et l'ensemble d'objets exprimé par : « *entre le Louvre et le Grévin* ». La solution retenue dans LOGUS-II pour représenter ce type d'énoncé est de définir des prédicats d'arité deux pour exprimer la nature de l'énoncé. Ainsi, la représentation sémantique du deuxième énoncé est :

$$\begin{aligned} ((\textit{interrogation le_quel}) \quad & (\textit{avec} (\textit{fermeture} [(\textit{horaire} (\textit{le_maxi tard}))]))) \\ & (\textit{entre} \quad (\textit{musee} [(\textit{identite} (\textit{nom} \textit{" Louvre"}))])) \\ & \quad (\textit{musee} [(\textit{identite} (\textit{nom} \textit{" Grevin"}))])) \end{aligned}$$

Dans cet exemple, le prédicat (*interrogation le_quel*) est d'arité deux : il prend en arguments une propriété et une chaîne d'objets.

On peut considérer cet exemple comme marginal. Cependant, il illustre la nécessité de pouvoir accroître le pouvoir d'expression du langage cible choisi. En effet, comme les structures sémantiques des énoncés des corpus utilisés sont souvent très simples, on peut craindre qu'elles ne reflètent que très partiellement les modes de l'expression orale spontanée d'utilisateurs qui auraient une assez grande confiance dans les facultés de compréhension de la machine à laquelle ils s'adressent. Les structures assez souples mises en place permettent cependant de faire face à ce type de besoin.

D'ores et déjà, on peut avancer que les principes adoptés d'une représentation sémantique obtenue par assemblage de concepts offre un pouvoir d'expression que l'on peut difficilement obtenir à l'aide de schémas sémantiques prédéfinis. Ces structures prédéfinies sont assez faciles à définir pour des tâches simples dans des domaines étroits : par exemple, la compréhension faite dans le système du LIMSI-ATIS sur les horaires de train repose sur la détection de l'une des cinq catégories de requêtes suivantes : *flight* (*vol*), *airfare* (*prix du billet*), *stop* (*escale*), *aircraft* (*type de l'avion*), *book* (*réservation*) (Minker, 1999; Minker, 1996). Il va de soi qu'un élargissement du domaine nécessite leur multiplication.

Par ailleurs, une structure sémantique modulaire telle que celle qui est proposée offre également l'avantage d'une modélisation facile de structures relativement complexes ; ces structures posent également problème dans le cas d'une représentation à l'aide de schémas sémantiques prédéterminés.

5.2.7 Conclusion

Le langage qui vient d'être défini utilise le caractère finalisé du dialogue : il repose sur un petit nombre de concepts qui représentent les objets du micro-monde concerné et les actes de

langage que l'on peut raisonnablement attendre dans un dialogue dont le but final correspond aux tâches envisagées. Son pouvoir d'expression est donc relativement élevé mais néanmoins limité au domaine de l'application : il correspond à la représentation du sens dit *utile* (cf. page 63).

Un analyseur tel que TINA (cf. 1.4.3), fournit au module de dialogue un arbre qui représente les dépendances syntaxico-sémantiques entre les différents éléments de l'énoncé ; on peut supposer que cette représentation doit ensuite être interprétée dans le langage de chacune des applications correspondantes. Dans LOGUS, l'analyse d'un énoncé consiste à traduire un énoncé de la langue naturelle dans le langage formel qui vient d'être présenté ; il s'agit d'une représentation sémantique au vrai sens du terme, sans traces des indices syntaxiques et sémantiques qui ont servi à la construire. Ces indices peuvent accessoirement être transmis au dialogueur, mais comme éléments donnés en parallèle, tels une notice indicative.

5.3 Les principes initiaux de l'analyse

Les structures de requêtes prédéfinies souvent utilisées dans les systèmes d'analyse de l'oral spontané constituent un cadre sémantique très contraignant. Elles définissent la langue cible : la représentation sémantique de l'énoncé correspond à la totalité ou à une partie de ces structures. Elles assurent donc une partie du rôle structurant de la syntaxe et permettent, dans les analyses sélectives (cf. §2.4.2), de limiter l'utilisation de celle-ci à des repérages de marques syntaxiques locales (déterminants, prépositions, etc.). Leur abandon nécessite donc de repenser les stratégies d'analyse adoptées.

L'analyse respecte le principe de composition de Frege selon lequel

« *la formule finale qui représente la compréhension de l'énoncé est une fonction de la signification de ses constituants et de la façon dont ils sont combinés* ».

Le problème est de déterminer des critères sur lesquels s'appuyer pour savoir comment combiner ces constituants.

Le pouvoir expressif des langages sans syntaxe indiquent que des connaissances sémantiques seules sont insuffisantes pour dégager sans ambiguïté la structure d'un énoncé (cf. page 32). Pour une compréhension « précise » des énoncés, il semble donc difficile de ne pas faire appel à des traitements syntaxiques.

5.3.1 Grammaires de Montague, λ -termes et analyse partielle

Les grammaires de Montague constitue un formalisme intéressant puisqu'elles permettent de faire un lien direct entre l'analyse syntaxique d'un énoncé et la construction de sa représentation sémantique. Celle-ci s'obtient par composition de λ -termes typés (ces formalismes sont rapidement présentés dans l'annexe A).

Par exemple, P. Delsarte et A. Thayse les utilisent pour proposer la traduction d'un fragment de la langue française en une formule logique, pour l'interrogation d'une base de données (Delsarte and Thayse, 2001). Leur prototype permet de transformer un énoncé de la langue naturelle en une formule du λ -calcul typé. Le système qu'ils construisent se situe dans un contexte pragmatique délimité proche de celui qui nous concerne. Il permet l'analyse de nombreuses tournures syntaxiques de la langue française ; l'utilisation de logiques temporelles et intentionnelles permet de donner un

pouvoir d'expression important au langage cible utilisé.

Cependant, une première différence importante avec le problème qui nous intéresse vient de ce que les auteurs ne se placent pas a priori dans un contexte de dialogue. Ils peuvent donc faire abstraction d'un bon nombre d'actes de dialogue possibles et des réponses elliptiques ou référentielles.

Mais surtout, dans leur système, l'analyse syntaxique complète des énoncés reste un préalable à leur interprétation. En ce sens, bien que l'objectif des auteurs ressemble au nôtre et malgré l'intérêt de la traduction proposée, le système qu'ils décrivent ne répond pas aux nécessités d'une analyse pragmatique des énoncés de la langue naturelle spontanée, dans un réel contexte d'utilisation.

Si la forme des énoncés à analyser exclut de toute façon la perspective d'une analyse syntaxique complète, le rapide tour d'horizon du chapitre 2 a montré que les prétraitements destinés à modifier cette forme n'ont donné jusqu'à présent que des résultats décevants.

Par ailleurs le chapitre 4 a montré que l'oral n'est pas aggrammatical et que,

« *une fois mises en place des procédures pour rendre compte des modes de production de l'oral, on s'aperçoit que les instruments utilisés pour la syntaxe de l'écrit conviennent parfaitement à celle de l'oral, à condition de prévoir une grande souplesse d'articulation dans le domaine des relations entre lexique et grammaire, où se font la plus grande partie des innovations* » (Blanche-Benveniste, 2002).

Cette affirmation ne peut s'appliquer à la lettre aux énoncés reconnus transmis pour analyse au système, puisqu'aux spécificités de l'expression orale spontanée, viennent s'ajouter les inévitables erreurs de reconnaissance. Cependant, elle laisse espérer que des traitements syntaxiques sont néanmoins envisageables.

5.3.2 Analyses syntaxiques partielles

Depuis quelques années, des méthodes d'analyses syntaxiques partielles robustes (*shallow parsing*) ont été développées en TALN pour les textes écrits. Elles sont en particulier utilisées pour automatiser l'extraction d'informations des grandes quantités de données que représentent les textes électroniques (Aït-Mokhtar et al., 2002; Ciravegna and Lavelli, 2002). Cette tâche présente des exigences qui s'apparentent à celles que requièrent l'analyse de la parole spontanée dans un système de DOHM : elle doit être rapide et robuste ; il ne s'agit pas de vérifier la correction linguistique des énoncés analysés et il convient même d'en tolérer les extra-grammaticalités. Leur seul but est d'extraire du sens.

Le principe général mis en œuvre dans ce type d'analyse consiste à produire des constituants syntaxiques minimaux, en général non récursifs, désignés suivant différents termes : *clusters*, *chunks*, *segments* ou *chains*. Ces groupes syntaxiques sont constitués d'une tête lexicale à laquelle sont rattachés des termes dépendants locaux.

Ces traitements cumulent les deux avantages suivants :

1. Ils sont robustes : en effet, les liaisons locales qu'ils effectuent sont sans ambiguïté. Les problèmes posés par les rattachements multiples possibles sont repoussés à des phases ultérieures d'analyse, s'appuyant sur d'autres règles. Cette absence d'ambiguïté des dépendances effectuées fait que les constituants ainsi obtenus constituent des « îlots de certitudes ».
2. Ils sont efficaces : des formalismes simples, tels que des automates à états finis, suffisent à la modélisation des dépendances locales concernées. Les analyses sont souvent incrément-

tales, comme chez Abney (Abney, 1996), par la recherche dans un ordre donné des groupes nominaux, verbaux et adjectivaux.

Par ailleurs, Abney fait valoir que les structures syntaxiques minimales, qu'il appelle *chunks* et qu'il définit comme des unités syntagmatiques non-récurrentes, correspondent à une réalité linguistique et même prosodique (Abney, 1991).

Or, des arguments linguistiques plaident en faveur de la possibilité de pratiquer cette étape de segmentation sur les énoncés de l'oral spontané. En effet, comme on l'a vu au §4.2, les études sur le français parlé attestent que des groupes syntaxiques minimaux locaux sont en général assez bien respectés dans les réparations (cf. page 53). C'est le cas par exemple dans les syntagmes nominaux avec les reprises des prépositions et des déterminants comme dans « *pour la euh vers la station enfin euh vers la station de métro* » (Martinie, 2001). Ainsi, ces processus préservent la cohérence de structures syntaxiques locales.

Cette segmentation en *chunks* a ainsi été utilisée par K. Zechner pour l'analyse de dialogues oraux dans des domaines non contraints, afin d'en extraire les informations essentielles (Zechner, 1998; Zechner, 2001). Elle est également utilisée dans le système de compréhension de la parole spontanée ROMUS développé par J. Goulian (Goulian, 2002).

Ainsi, si une telle segmentation ne peut suffire à dégager entièrement le sens d'un énoncé, elle permet d'utiliser au mieux les indices syntaxiques locaux. De plus, contrairement aux travaux de nettoyage précédemment cités (cf. page 35), elle présente l'avantage essentiel d'être *non destructive* : comme elle se contente d'effectuer des regroupements, tous les éléments de l'énoncé restent utilisables pour la poursuite de l'analyse.

Leur efficacité et leur adéquation à la langue orale spontanée rendent a priori les analyses syntaxiques partielles particulièrement intéressantes : le premier principe d'analyse retenu dans LOGUS est donc celui d'une segmentation en *chunks*. Le formalisme utilisé pour cette étape est celui des grammaires catégorielles. Comme dans le calcul de Lambek⁴, chaque application des règles de grammaire détermine la composition des λ -termes qui représentent sémantiquement les éléments de l'énoncé.

Liens sémantiques

Quelle que soit son efficacité, la segmentation en *chunks* n'est qu'une étape dans la compréhension de l'énoncé. La construction de la représentation sémantique finale nécessite une étude des dépendances entre les constituants obtenus. Les réparations et la variabilité de l'organisation syntaxique des énoncés oraux rendent illusoire l'idée de fonder l'étude de ces dépendances sur la syntaxe seule. Le principe initialement retenu pour cette étape de l'analyse est de la faire reposer sur une étude sémantique.

Un domaine d'application relativement circonscrit et bien délimité permet de définir une connaissance sémantique qui lui est spécifique. Dans LOGUS, elle est envisagée comme une ontologie des liens sémantiques possibles entre les objets du domaine et leurs propriétés. Il ne s'agit pas de décrire des structures prédéfinies, mais de décrire la liste des rattachements sémantiquement possibles entre les différents éléments. Le principe essentiel retenu a priori pour les étapes suivantes est de privilégier les compositions qui « font sens » au sens d'une connaissance sémantique.

4. Les formalismes correspondants sont présentés dans l'annexe A.

tique définie sur les objets du domaine. Ce principe directeur garantit la cohérence sémantique des formules obtenues.

J.-P. Balpe définit le sens comme « *un tissage de relations* » ; il précise même qu’il s’agit selon lui, « *d’un tissage de relations effectives dans un ensemble de relations possibles* », d’« *une mise en contexte* » (Balpe, 2002). l’analyse faite dans LOGUS met en œuvre cette définition. Le langage cible peut alors être comparé à un jeu de construction ; les éléments en sont les pièces et les emboîtements possibles des différentes pièces entre elles sont définis dans la connaissance sémantique. La stratégie d’analyse la plus élémentaire consiste à considérer que la meilleure analyse est celle qui permet de relier le plus grand nombre possible des éléments signifiants de l’énoncé.

5.4 LOGUS-I : description et évaluations

LOGUS-I est une mise en œuvre des principes qui viennent d’être exposés⁵. La première partie de ce paragraphe est une description rapide de son architecture et des principales étapes de l’analyse. La seconde partie présente les conclusions des évaluations opérées sur ce premier système ; elle explique comment elles ont permis de juger de la validité de cette approche mais aussi de remettre en cause quelques-uns des principes adoptés et le formalisme choisi.

5.4.1 L’architecture de LOGUS-I

L’architecture générale de LOGUS-I est schématisée dans la figure 5.3.

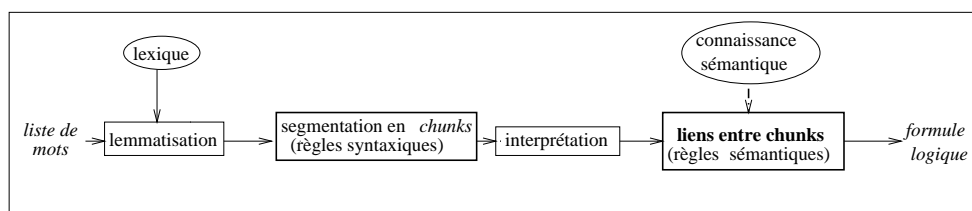


FIG. 5.3 – Architecture de LOGUS-I

Quatre traitements successifs sont appliqués (les deux traitements principaux sont en gras dans la figure) :

1. Lemmatisation

Chacun des lexèmes de la liste d’entrée présent dans le lexique est transformé en une liste de ses définitions.

Une définition est un couple : son premier élément est une étiquette syntaxique ; le second est un λ -terme qui doit servir à obtenir, par composition, la représentation sémantique de l’énoncé. Les constantes qui y figurent doivent être des éléments définis dans le langage cible (cf. chapitre 5.2).

Par exemple, le mot « *chambre* » admet pour définition (*nomc chambre*), où l’étiquette syntaxique *nomc* signifie nom commun. Pour les mots grammaticaux destinés à se rattacher à

5. Les langues cibles des deux versions de LOGUS, LOGUS-I et LOGUS-II présentent quelques différences. Les principes étant semblables, seule celle de LOGUS-II a été présentée.

une tête lexicale, l'étiquette syntaxique est fractionnaire et le λ -terme est une abstraction. Par exemple, l'une des définitions du mot « *une* » est ($c_nom/nomc \lambda x.(x (det\ ind\ef\ sing))$); $c_nom/nomc$ est l'étiquette syntaxique d'un élément qui peut se combiner avec un nom commun (*nomc*) situé à sa droite pour former un *chunk* nominal (c_nom), le λ -terme permet la composition avec ce nom commun suivant les règles qui vont être décrites.

2. Segmentation en *chunks*

La segmentation en *chunks* permet d'associer des mots juxtaposés sur la base de règles exclusivement syntaxiques. Comme il a été dit (cf. page 77), un formalisme simple suffit pour modéliser les dépendances locales concernées : deux règles dérivées de celles des grammaires catégorielles⁶ de type AB (Bar-Hillel, 1964) sont appliquées aux étiquettes syntaxiques :

- $(A, A \setminus B) \rightarrow B$: composition d'un élément de catégorie A avec un élément de catégorie fractionnaire $A \setminus B$ situé à sa droite pour former un élément de catégorie B .
- $(B/A, A) \rightarrow B$: composition d'un élément de catégorie fractionnaire B/A avec un élément de catégorie A situé à sa droite pour former un élément de catégorie B .

Le λ -terme qui détermine la partie sémantique du couple résultat s'obtient par composition des λ -termes correspondants ; Par exemple, le λ -terme correspondant au *chunk* adjectival « *pas trop cher* » s'obtient comme composition des λ -termes respectivement associés aux trois mots qui le composent :

lexème	catégorie syntaxique	λ -terme
« <i>pas</i> »	adj/adj	$\lambda x.(pas\ x)$
« <i>trop</i> »	adj/adj	$\lambda x.x$
« <i>cher</i> »	adj	<i>cher</i>
« <i>pas trop cher</i> »	adj	$(\lambda x.(pas\ x) (\lambda x.(x\ cher)))$ $\equiv_{\beta} (pas\ cher)$

3. Interprétation et filtrage

Les *chunks* ainsi obtenus sont d'abord « interprétés⁷ », c'est à dire transformés en « éléments sémantiques » de trois types : les deux premiers correspondent aux objets et à leurs propriétés ; le troisième contient les mots qui indiquent la structure de l'énoncé (coordinations, négations, etc.) et les groupes de mots qui permettent de connaître la nature de l'énoncé (appelés « mots-questions » et « mots-requêtes »). Par exemple, le constituant ($adj (pas\ cher)$) est interprété comme une propriété éventuelle d'un objet du domaine ($prop\ tarif (pas\ cher)$) : propriété d'étiquette *tarif* et de détermination (*pas cher*) (cf. 5.2.2).

Les autres constituants, considérés comme non significatifs, sont éliminés ; à ce stade, disparaissent ainsi les mots inconnus⁸ d'une part et d'autre part les mots grammaticaux non rattachés tels que les déterminants et les prépositions, amorces de groupes nominaux dans les réparations.

4. Liens entre *chunks*

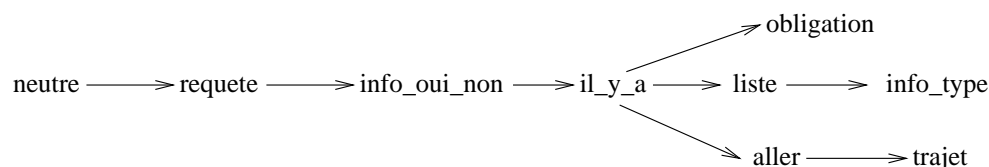
Le quatrième et dernier traitement consiste à lier entre eux les constituants sémantiquement interprétés pour obtenir la formule logique finale. Des prédicats logiques décrivent les liens

6. Ce formalisme est décrit dans le paragraphe A.1 de l'annexe A.

7. Il s'agit d'une interprétation linguistique, qui repose partiellement sur le contexte du domaine de l'application, mais ni sur le contexte du dialogue, ni sur le contexte formé par les autres éléments de l'énoncé.

8. Le problème des mots inconnus et de leur traitement est repris au 7.2.1. Le terme de *mots inconnus* ne désigne ici que les mots absents du lexique de LOGUS-I.

possibles entre les objets et les propriétés. Ils définissent ainsi une connaissance sémantique spécifique du domaine. Les règles utilisées sont exclusivement sémantiques : la stratégie définie dans LOGUS-I consiste à fabriquer des « chaînes d'objets » maximales, c'est à dire qui intègrent le plus grand nombre possible des objets et propriétés présents dans l'énoncé, en utilisant les liens décrits dans la connaissance sémantique. La détermination de l'acte de langage (ou tout au moins des indices linguistiques présents dans l'énoncé sur la nature de cet acte) s'obtiennent en utilisant une relation d'ordre sur les mots-questions et mots-requêtes⁹ :



Par exemple, une expression telle que « *je voudrais savoir s'il y a* » est, à l'issue de l'étape d'interprétation, modélisée par la succession de trois constituants *requete*, *info_oui_non* et *il_y_a*. La relation d'ordre indique que ces trois constituants peuvent se réduire au dernier d'entre eux : *il_y_a*¹⁰.

5.4.2 Évaluation de LOGUS-I et conclusions

Les difficultés que présente l'évaluation des systèmes de compréhension ont fait l'objet d'un paragraphe du chapitre 1 (cf. page 19). LOGUS-I a participé à la campagne d'évaluation dite « par défi » (Antoine et al., 2002a) dans le cadre du groupe de travail 5.1 « compréhension robuste » du GDR-I3 du CNRS dont le paradigme a été présenté dans ce même paragraphe (page 23).

LOGUS-I a ainsi été soumis à 1200 énoncés tests répartis suivant quatre séries de 300 énoncés très différentes. Bien que simulés, ces tests, qui représentent une sorte de catalogue des difficultés rencontrées par tous les participants, ont été révélateurs des comportements des systèmes testés.

Même si la campagne d'évaluation « par défi » n'a pas conduit à des conclusions objectives permettant une comparaison directe de LOGUS-I avec les autres systèmes testés, on peut quand même avancer que face aux autres systèmes, LOGUS-I a semblé avoir dans l'ensemble un comportement satisfaisant.

Les résultats obtenus (des données chiffrées figurent dans le paragraphe 7.1, page 115 où sont comparés les performances de LOGUS-I et de LOGUS-II) montrent une grande robustesse du système face à l'ordre des différents *chunks* (environ 10% d'erreurs), à la plupart des extra-grammaticalités dues aux modes de production de l'expression orale : corrections, répétitions, reprises (5% d'erreurs environ), et aux objets complexes. Entre autres, peu d'erreurs sont engendrées par l'étape de la segmentation en *chunks* et les liens sémantiques établis entre les objets et leurs propriétés sont très rarement erronés. Ainsi, les principes utilisés dans le prototype semblent dans une certaine mesure justifiés :

- Une segmentation en *chunks* semble possible et efficace pour les énoncés oraux. Cependant, cette segmentation peut engendrer des erreurs lorsque des « réparations » ou ellipses

9. Certains concepts considérés comme des éléments de questions ou de requêtes dans LOGUS-I (*aller*, *trajet*, etc.) ont été intégrés au monde des objets par la suite.

10. Pour plus de détails, on peut se reporter à (Villaneau et al., 2001).

s'opèrent à l'intérieur d'un *chunk*. Par exemple, la segmentation « [dans un deux] [ou] [dans un trois étoiles] » entraîne soit une perte d'information (si le groupe de mots [« dans un deux »] est négligé faute de contenu sémantique « suffisant ») soit la nécessité de définir des coordinations complexes (les deux éléments reliés par la coordination « ou » sont de natures différentes).

- Par ailleurs, le très faible pourcentage de liens erronés entre objets et propriétés valident les principes qui président à la construction des chaînes d'objets. L'abandon de schémas sémantiques prédéterminés peut faire croire au risque de l'établissement de liens erronés : la base de connaissance des dépendances sémantiques semble suffire pour y remédier.

Cependant, une analyse qualitative des résultats montre également les limites de l'approche et du formalisme :

- L'absence totale de syntaxe dans la deuxième partie de l'analyse rend difficile la reconnaissance des faux-départs (« *quels sont les horaires enfin non d'abord je voudrais les tarifs...* ») et des incisives. Cette même absence de syntaxe ne permet pas de distinguer les différentes parties d'un énoncé lorsqu'elles correspondent à la succession de deux ou plusieurs actes de langage ; c'est le cas par exemple des énoncés du type information-requête, absent des tests de l'évaluation mais assez fréquemment rencontré dans les corpus, tel que, par exemple : « *j'ai réservé à l'hôtel Caumartin comment je peux faire pour y aller d'ici* ». Dans chacun de ces deux exemples, une identification même grossière des groupes verbaux rendrait cette distinction possible.
- Le lexique se restreint aux mots jugés « utiles » à la compréhension et à quelques mots courants jugés « vides » d'un point de vue sémantique. Les mots absents du vocabulaire de LOGUS, qualifiés d' « inconnus » (cf. note 3, page 80), sont éliminés à la fin de la segmentation en *chunks*. Cette élimination pure et simple aussi précoce accentue le problème de la reconnaissance des faux-départs et des incisives.
- Par ailleurs, ce prototype est trop dépendant de l'application étudiée. L'étape d'interprétation qui suit la segmentation en *chunks* est directement liée au domaine de l'application.. Par ailleurs, les règles utilisées dans la phase d'établissement des liens sémantiques dépendent directement des objets du domaine et, par conséquent, manquent de généralité.

Le chapitre suivant montre comment ces différents défauts ont été corrigés dans LOGUS-II, la version suivante de LOGUS.

Chapitre 6

Seconde mise en œuvre

Ce chapitre est consacré à la description de l'analyse faite dans LOGUS pour traduire dans la langue cible les énoncés soumis à la compréhension. La première partie présente les principes et le formalisme utilisés. La suite du chapitre décrit les différentes étapes de l'analyse.

6.1 LOGUS-II : principes et formalisme

6.1.1 Les principes

Trois principes directeurs président à la conception de LOGUS-II, la deuxième version de LOGUS :

1. Le premier principe consiste à faire débiter l'analyse par une étape de segmentation en *chunks* : les résultats de l'évaluation subie par LOGUS-I ont en effet montré que ce type de traitement conserve sa qualité de robustesse lorsqu'on le pratique sur les énoncés de l'oral spontané. Les quelques erreurs générées ont cependant fait voir la nécessité d'adapter la notion de *chunk* à l'analyse de ce type d'énoncés.

La segmentation en *chunks* recouvre des pratiques très différentes mais, au sens utilisé par exemple par Abney (Abney, 1991), l'expression « *une chambre double* » constitue un seul *chunk* puisqu'elle constitue un unique groupe nominal. Ainsi, l'adjectif « *double* » est rattaché au nom commun « *chambre* » considéré comme sa tête lexicale. Un tel rattachement pose problème pour l'analyse de certains énoncés oraux ; il rend en effet plus difficile l'analyse d'expressions telles que : « *une chambre double ou bien triple* », « *une chambre double non pardon triple plutôt* », etc. Les coordinations sont en effet plus faciles à définir entre des éléments syntaxiquement et sémantiquement similaires *double/triple*, qu'entre des éléments qui ne le sont pas (*une chambre double*)/*triple*. Ainsi donc, dans LOGUS, un adjectif suivant un *chunk* nominal constitue un *chunk* à part entière.

Si l'on se réfère encore une fois à la notion de *chunk* proposée par Abney comme unité syntaxique et prosodique, il semble qu'une suite de mots de la forme (*déterminant, adjectif, nom commun*) doive constituer un seul *chunk*. Pourtant ce type de rattachement pose des problèmes analogues à ceux du cas précédent pour l'analyse d'expressions du type : « *dans un grand ou même pourquoi pas un très grand hôtel* », « *dans un deux ou dans un trois étoiles* », etc. En effet, si l'on regroupe dans un même *chunk* « *dans un trois*

étoiles » et si l'on ne veut pas que le début de l'expression soit laissé pour compte, on est amené à définir « *dans un deux* » comme un *chunk* d'un type syntaxique particulier (*pre_nom_commun_prepositionnel*); ce choix nécessite alors de prévoir dans les règles qui établissent les dépendances entre *chunks*, une coordination possible entre ce type de *chunk* et le *chunk* nominal « *dans un trois étoiles* ». Il est alors plus simple d'attribuer le même statut à la suite « *dans un trois* », de découper l'expression suivant « *[dans un deux] [ou] [dans un trois] [étoiles]* ». La règle de coordination à mettre en œuvre lors de l'analyse des dépendance entre *chunks* est alors beaucoup plus naturelle.

Un autre problème lié au rattachement des objets et propriétés dès l'étape de segmentation en *chunks* est posé par l'ambiguïté sémantique. Par exemple, le mot « *grand* » correspond à des propriétés différentes dans les deux expressions « *un grand restaurant* » et « *une grande chambre* ». Avec l'approche adoptée dans LOGUS, c'est la base de connaissance des dépendances syntaxico-sémantiques entre objets et propriétés qui permet de lever cette ambiguïté. Pour éliminer ces deux types de problèmes, la solution adoptée dans LOGUS-II consiste à restreindre la notion de *chunks* en adoptant les définitions et la règle suivante :

Définition 1 *Un lexème est dit lexical si son composant sémantique contient un objet, une propriété ou l'expression d'une requête définis dans l'application.*

Définition 2 *Un chunk minimal contient au plus un élément lexical.*

Ces définitions modifient la notion même de *chunk*; un *chunk* est en effet généralement conçu comme étant défini sur la base de critères exclusivement syntaxiques. Or, la définition qui vient d'être donnée de *chunk minimal* fait intervenir un critère sémantique. D'une manière générale, la stratégie d'analyse utilisée dans LOGUS-II fait s'interpénétrer les arguments syntaxiques et sémantiques à tous les niveaux.

Règle 1 *La segmentation en chunks ne doit construire que des chunks minimaux.*

L'adoption de cette règle conduit par exemple aux segmentations suivantes :

[« *dans un deux* »] [« *ou* »] [« *dans un trois* »] [« *étoiles* »],
[« *un grand* »] [« *restaurant* »].

La première de ces deux segmentations permet un traitement simple de la coordination et la seconde une interprétation non ambiguë de la propriété portée par l'adjectif.

2. La seconde idée directrice consiste à se donner la possibilité d'utiliser des arguments syntaxiques pour étudier les dépendances entre *chunks*. Même si cette absence de syntaxe dans LOGUS-I n'a pas provoqué un grand nombre d'erreurs de compréhension lors de l'évaluation, elle semble restreindre les possibilités réelles de l'approche.
3. Le troisième objectif est de parvenir à rendre génériques les règles utilisées. L'une des qualités recherchées pour LOGUS est la facilité à pouvoir adapter le système à d'autres applications. LOGUS-I ne répond pas assez bien à ce principe fondamental (cf. page 82).

Le formalisme qui est présenté dans la suite de ce paragraphe est un réponse aux deux derniers des trois objectifs présentés : il permet en effet la suppression de l'étape d'interprétation qui suit la segmentation en *chunks* dans LOGUS-I et l'utilisation conjointe d'arguments syntaxiques et sémantiques pour l'étude des dépendances entre ces *chunks*.

6.1.2 Le formalisme de représentation des éléments

Dans LOGUS-II, les constituants de l'énoncé sont représentés par des triplets tout au long de l'analyse, à partir des définitions données par le lexique jusqu'à la fin des étapes de composition. Les différents éléments d'un triplet $\langle C, R, T \rangle$ sont les suivants :

1. Le premier élément C est appelé *catégorie syntaxique*. Il joue le rôle d'une étiquette syntaxique. Lors de la segmentation en *chunks*, il intervient en tant que *catégorie* au sens des grammaires catégorielles. Au niveau de l'étude des dépendances entre *chunks*, il permet d'imposer des contraintes syntaxiques dans les règles de composition.
2. Le second élément R est appelé *rôle sémantique*. Il s'agit d'un étiquetage sémantique des éléments qui indique la fonction sémantique qu'ils occupent dans le langage cible : objets, propriété, etc.
3. Le troisième élément T est la *traduction sémantique*, traduction du constituant dans le langage cible ; elle appartient donc au langage de représentation défini précédemment (page 62). Elle se construit par compositions successives, au fur et à mesure du déroulement de l'analyse.

Les catégories syntaxiques

La plupart des catégories syntaxiques *simples* présentes dans les définitions du lexique ou obtenues à l'issue de la segmentation en *chunks* sont directement inspirées des étiquettes syntaxiques classiques. Elles peuvent être ou non munies d'attributs. Par exemple, la catégorie syntaxique d'un déterminant comporte en attributs la nature du déterminant (possessif, défini, indéfini, etc) et le nombre¹ ; la catégorie syntaxique de l'expression « *une chambre* » est ($c_nom\ nomc\ (det\ def\ sing)$) pour indiquer qu'il s'agit d'un *chunk* nominal (c_nom) dont la tête est un nom commun ($nomc$) auquel est rattaché un déterminant défini singulier ($(det\ def\ sing)$).

Cependant, un certain nombre d'entre elles ont été élaborées afin de faciliter la définition des règles syntaxico-sémantiques utilisées pour composer les *chunks*. En général, elles correspondent à des spécifications syntaxiques plus précises que les catégories syntaxiques habituelles ; c'est ainsi par exemple qu'à l'issue de la segmentation, un *chunk* tel que « *vers le plus proche* » se voit attribuer la catégorie syntaxique ($superlatif\ sing\ (preposition\ ad)$) avec deux attributs qui précisent respectivement le nombre ($sing$) et la préposition ($preposition\ ad$).

L'étude des dépendances entre *chunks* conduit à la construction de chaînes d'objets. Ces chaînes sont obtenues par application de la *relation d'enchaînement* ou d'une coordination entre objets (un élément de $Coord_obj$) (cf . page 64). Pour la poursuite de la construction de la représentation sémantique, il peut être utile de pouvoir disposer de la nature syntaxique de chacun des éléments de la chaîne. Les deux constructeurs $categs$ et $conj$ permettent de créer des catégories syntaxiques complexes pour ce type de structure, où la catégorie syntaxique des différents

1. Les erreurs de reconnaissance sur les mots courts, les erreurs fréquentes de genre faites par les locuteurs eux-mêmes et le fait que les formes fléchies des adjectifs ou des participes passés féminines et masculines sont souvent homophones nous ont conduit à ne pas tenir compte du genre. Par contre, nous faisons la distinction entre singulier et pluriel. Ce choix est bien entendu très discutable et probablement provisoire : l'étude des références risque en effet de rendre utile l'utilisation du genre (sa prise en compte ne pose aucun problème). Il convient de préciser cependant que l'utilisation des indications de nombre ne sont utilisées que comme indice de priorité entre rattachements : elles privilégient certains d'entre eux dans les cas d'ambiguïté.

constituants reste accessible. Par exemple, si C_1 et C_2 sont les catégories syntaxiques respectivement associées à l'objet simple O_1 et à la chaîne O_2 , alors, la catégorie syntaxique de $(de\ O_1\ O_2)$ est $(categs\ C_1\ C_2)$; de même, si C_1 et C_2 sont les catégories syntaxiques respectivement associées à la chaîne O_1 et à la chaîne O_2 et si $coord \in Coord_obj$, alors la catégorie syntaxique de $(coord\ O_1\ O_2)$ est $(conj\ C_1\ C_2\ coord)$.

Formellement, ces constructeurs sont simplement définis de la façon suivante :

Définitions :

1. Si C_1 et C_2 sont des catégories syntaxiques, alors $(categs\ C_1\ C_2)$ est une catégorie syntaxique.
2. Si C_1 et C_2 sont des catégories syntaxiques et si $coord \in Coord_obj$, alors $(conj\ C_1\ C_2\ coord)$ est une catégorie syntaxique.

Exemples

1. Par exemple, la chaîne d'objet qui correspond à l'expression « réserver une chambre double au Caumartin » est

$$(de\ (reservation\ [])\ (de\ (chambre\ [(taille\ double)])\ (hotel\ [(identite\ (nom\ "Caumartin"))]))).$$

Elle se voit attribuer la catégorie syntaxique

$$(categs\ infinitif\ (categs\ (c_nom\ nomc\ (det\ indef\ sing))\ (c_nom_p\ nomp\ (preposition\ a_prep)))).$$

2. Par exemple, la chaîne qui correspond à « les tarifs du Louvre et ses horaires » est :

$$(et\ (de\ (tarif\ [])\ (musee\ [(identite\ (nom\ "Louvre"))]))\ (horaire\ []))$$

de catégorie syntaxique

$$(conj\ (categs\ (c_nom\ nomc\ (det\ def\ plur))\ (c_nom\ nomp\ (preposition\ de_prep)))\ (c_nom\ nomc\ (det\ poss\ plur))\ et)$$

Le tableau de la figure 6.1 énumère et décrit la plupart des catégories syntaxiques utilisées dans LOGUS-II avec leurs attributs (ces attributs sont précédés de *). Celui de la figure 6.2 donne des exemples relatifs à ces catégories : le nombre qui figure dans la colonne de gauche renvoie à celui indiqué dans la colonne correspondante du tableau précédent.

Les rôles sémantiques

Les rôles sémantiques ont une fonction équivalente à celle des étiquettes syntaxiques : permettre l'identification de la fonction sémantico-pragmatique d'un élément.

	Catégorie	Définition
1	(<i>det *nature *nombre</i>)	déterminant : * <i>nature</i> : défini, possessif, etc. * <i>nombre</i> : singulier, pluriel
2	(<i>preposition *nature_prep</i>)	préposition * <i>nature_prep</i> : ad,in,ab,etc.
3	<i>nomc</i>	nom commun
4	<i>nomp</i>	nom propre
5	(<i>c_nom *categ *determinant</i>)	<i>chunk</i> nominal * <i>categ</i> : catégorie de la tête lexicale
6	(<i>c_nom_p *categ *prepos.</i>)	<i>chunk</i> nominal prépositionnel
7	(<i>pronom *nature *pers *nombre</i>)	les pronoms * <i>pers</i> : 1 ^{iere} , 2 ^{ieme} ou 3 ^{ieme} personne
8	(<i>pre_c_nom *categ</i>)	<i>chunk</i> lexical début d'un <i>chunk</i> nominal (cf. principe 1, page 83)
9	(<i>pre_c_nom_p *categ *prepos.</i>)	<i>idem</i> précédé d'une préposition
10	(<i>adj_num *ordre</i>)	adjectifs numériques * <i>ordre</i> : ordre de grandeur
11	<i>adjectif</i>	adjectifs
12	<i>c_adj</i>	<i>chunk</i> adjectival
13	<i>adjectif_pre_pose</i>	adjectifs placés avant le nom qu'ils déterminent
14	<i>adverbe</i>	adverbes
15	(<i>superlatif *nombre *nature_prep</i>)	les superlatifs
16	(<i>comparatif *comp. *categ</i>)	les comparatifs
17	<i>expr</i>	expressions diverses
18	<i>infinitif</i>	infinitif
19	(<i>infinitif_p *categ</i>)	infinitif introduits par préposition
20	(<i>pp *aux</i>)	participe passé * <i>aux</i> : auxiliaire qu'il gouverne
21	(<i>verbe *pers *temps</i>)	verbe conjugué * <i>temps</i> : passé, présent, futur
22	(<i>gv *pers *temps</i>)	sujet + verbe
23	<i>interjection, conjonction, particule</i>	interjections, conjonctions et particules
24	(<i>categs C₁ C₂</i>)	catégorie syntaxique résultat de l'association de deux constituants de catégories resp. <i>C₁</i> et <i>C₂</i>
25	(<i>conj C₁ C₂ coord</i>)	catégorie syntaxique résultat de la coordination de deux constituants de catégories resp. <i>C₁</i> et <i>C₂</i>

FIG. 6.1 – Les principales catégories syntaxiques utilisées dans LOGUS-II

Exemple		Catégorie syntaxique
1	« le »	(<i>det def sing</i>)
2	« dans »	(<i>preposition in</i>)
3	« chambre »	<i>nomc</i>
4	« Louvre »	<i>nomp</i>
5	« une chambre »	(<i>c_nom nomc (det indef sing)</i>)
6	« vers la gare »	(<i>c_nom_p (c_nom nomc (det def sing)</i> (<i>preposition ad</i>))
	« en chambre »	(<i>c_nom_p nomc (preposition ad)</i>)
7	« celui »	(<i>pronom indef 3 sing</i>)
8	« un grand »	(<i>pre_c_nom adjectif_pre_pose (det indef sing)</i>)
9	« dans un deux »	(<i>pre_c_nom_p (pre_c_nom (adj_num "inf10")</i> (<i>det indef sing</i>)) (<i>preposition in</i>))
12	« pas trop cher »	<i>c_adj</i>
15	« dans le plus cher »	(<i>superlatif sing (preposition in)</i>)
16	« moins d'étoiles »	(<i>comparatif moins_que</i> (<i>c_nom_p (preposition de_prep)</i>)
19	« pour aller »	(<i>infinitif_p (preposition pour)</i>)
22	« j'ai réservé »	(<i>gv 1 passe</i>)
24	« du deux au trois »	(<i>catégs</i> (<i>pre_c_nom_p (pre_c_nom (adj_num "inf10")</i> (<i>det indef sing</i>)) (<i>preposition de</i>)) (<i>pre_c_nom_p (pre_c_nom (adj_num "inf10")</i> (<i>det indef sing</i>)) (<i>preposition a</i>)))
25	« deux ou trois »	(<i>conj (adj_num "inf10") (adj_num "inf10") ou</i>)

FIG. 6.2 – Exemples de catégories syntaxiques

L'un des buts à atteindre est la généralité des règles syntaxico-sémantiques qui président à la composition des *chunks*. Ces règles utilisent les deux premiers champs du triplet qui représentent les éléments de l'énoncé : catégorie syntaxique et rôle sémantique. Pour qu'elles soient aussi générales que possible, il convient donc de déterminer une liste de rôles sémantiques qui soient autant que possible indépendants du domaine de l'application concernée. On peut classer les rôles qui ont été choisis dans LOGUS suivant cinq classes différentes, auxquelles on peut adjoindre la classe des constructeurs de rôles. Le tableau de la figure 6.3 donne la liste des principaux rôles sémantiques utilisés ; celui de la figure 6.4 donne des exemples de ces rôles pour chacune des classes présentées ci-dessous.

1. Les objets ou actions de l'application sont regroupés sous le rôle sémantique générique de *obj_acte*.
2. Les propriétés ont pour rôle sémantique (*prop R*) où l'attribut *R* correspond aux étiquettes des propriétés telles qu'elles ont été définies au paragraphe 5.2.2. La liste de ces étiquettes est largement indépendante de l'application mais on ne peut pas prétendre cependant qu'elle le soit totalement : par exemple l'étiquette de propriété *nbetoiles* est relative aux étoiles des hôtels. De même, l'étiquette *prix* n'existe que parce que certains objets du domaine sont payants ; elle peut par contre être utilisée dans tout autre domaine où de tels objets existent.
3. Certains rôles désignent des éléments qui donnent des indices sur les intentions du locuteur : ce peuvent être des interrogations marquées par des mots interrogatifs, des requêtes non interrogatives (« *je voudrais* »), des demandes explicites (« *réservez-moi* »), des modalités (« *c'est possible* »), etc.
4. Un certain nombre de rôles sont attribués à des éléments de l'énoncé qui permettent de contrôler certaines compositions, tels que les coordinations, les marques de reprise ou d'hésitation.
5. Enfin, d'autres rôles servent à indiquer le caractère non essentiel d'un élément.
6. Les rôles incomplets correspondent au rôle attribué à un élément qui, associé avec un autre, permettra d'obtenir un rôle donné. Leur utilité est liée au caractère minimaliste des *chunks* : ils sont destinés à faciliter l'association entre certains d'entre eux.
Par exemple, l'expression « *deux étoiles* » correspond à deux *chunks minimaux*. L'une des définitions associée au *chunk étoiles* a pour rôle sémantique (*unite (prop nbetoiles)*). Ce rôle indique une association prioritaire de ce *chunk* avec un *chunk* de rôle sémantique (*prop entier*) pour obtenir un constituant de rôle (*prop nbetoiles*).
7. Association de rôles : l'application d'un acte de langage à une chaîne d'objets, la coordination de deux propriétés correspondent à l'association de deux constituants de rôles différents ; le constructeur *roles* permet d'en rendre compte.

Définition : Si R_1 et R_2 sont des rôles sémantiques, alors (*roles* R_1 R_2) est un rôle sémantique.

Exemples :

- (a) Le rôle sémantique de l'expression « *quels sont les hôtels* » est (*roles interrogation obj_acte*).
- (b) Le rôle sémantique de l'expression « *deux étoiles et pas trop cher* » est (*roles (prop nbetoiles) (prop prix)*).

Rôles sémantiques		Définition
1	<i>obj_acte</i>	- objets et actions de l'application
2	<i>(prop *R)</i>	- propriétés *R : étiquette de la propriété
3	<i>interrogation</i> <i>requete</i> <i>demande</i> <i>information</i>	- forme interrogative (partielle cf. 55) - marque de requête - marque d'un ordre (location ou réservation) - marque de la donnée d'une information
4	<i>(coordination *C)</i> <i>reprise</i> <i>excuse</i> <i>hesitation</i> <i>(auxiliaire *N)</i>	- mots de coordination *C : nature de la coordination - marque de reprise - marque d'excuse - marque d'hésitation - les verbes auxiliaires
5	<i>annexe</i> <i>neglige</i> <i>r_inconnu</i>	- pas de rôle sémantique précis - absence a priori de rôle sémantique - rôle des mots absents du lexique
6	<i>(unite R)</i> <i>(preobj R LC)</i>	- permet d'obtenir un élément de rôle <i>R</i> lorsqu'on l'associe à un nombre - permet d'obtenir un élément de rôle <i>R</i> lié à un objet qui satisfait des conditions syntaxiques données dans <i>LC</i>
7	<i>(roles R₁ R₂)</i>	- association des rôles <i>R₁</i> et <i>R₂</i>

FIG. 6.3 – Les principaux rôles sémantiques définis dans LOGUS-II

Les traductions sémantiques

Si un constituant contient des éléments lexicaux au sens défini précédemment (cf. page 84), sa *traduction sémantique* est sa traduction dans la langue cible. Les autres constituants (marques de coordinations, etc) et les mots hors-vocabulaire se voient attribuer la traduction sémantique générique de *neutre*.

Lorsque deux éléments de l'énoncé sont assemblés, la traduction sémantique résultante s'obtient essentiellement à partir des traductions sémantique des deux éléments concernés mais peut également faire intervenir leurs rôles sémantiques. Par exemple, lors du rattachement des éléments « un hôtel » et « deux étoiles », l'étiquette de la propriété intervient pour la construction de la traduction sémantique de l'élément final.

Par exemple :

« un hôtel » : $\langle (c_nom\ nomc\ (det\ indef\ sing)),\ obj_acte,\ (hotel\ []) \rangle$
« deux étoiles » : $\langle (c_nom\ nomc\ (adj_num\ "inf10")), (prop\ nbetoiles), (entier\ 2) \rangle$

« un hôtel deux étoiles »

$\langle (c_nom\ nomc\ (det\ indef\ sing)),\ obj_acte,\ (hotel\ [(nbetoiles\ (entier\ 2))]) \rangle$

	Exemple	Rôle sémantique
1	« <i>une chambre</i> » « <i>réserver</i> » « <i>le prix d'un ticket de métro</i> »	<i>obj_acte</i> <i>obj_acte</i> <i>obj_acte</i>
2	« <i>le plus cher</i> » « <i>dans un deux</i> » « <i>samedi</i> »	(<i>prop tarif</i>) (<i>prop ent</i>) (<i>prop nom_jour</i>)
3	« <i>quel</i> » « <i>je voudrais</i> » « <i>possible</i> »	<i>interrogation</i> <i>requete</i> <i>modalite</i>
4	« <i>je veux dire</i> » « <i>et puis aussi</i> »	<i>reprise</i> (<i>coordination "et"</i>)
5	« <i>dans</i> » « <i>en fait</i> »	<i>annexe</i> <i>neglige</i>
6	« <i>étoiles</i> » « <i>proche</i> »	(<i>unite (prop nbetoiles)</i>) (<i>preobj (prop proche_de) [(preposition de)]</i>)
7	« <i>j'ai réservé</i> » « <i>quels sont les hôtels</i> »	(<i>roles information obj_acte</i>) (<i>roles interrogation obj_acte</i>)

FIG. 6.4 – *Quelques exemples de rôles sémantiques*

Intérêt du formalisme

Le formalisme de représentation des éléments de l'énoncé tel qu'il vient d'être proposé offre plusieurs avantages :

1. Le premier avantage est qu'il peut être utilisé d'un bout à l'autre de l'analyse. Dans LOGUS-I, la segmentation en *chunks* et l'étude des dépendances entre ces *chunks* utilisent des formalismes différents ; l'étape d'interprétation qui sert à passer du premier formalisme au second présente l'inconvénient d'être très étroitement liée au domaine de l'application et de pouvoir difficilement être réalisée par des règles génériques. Cette étape n'est pas nécessaire dans LOGUS-II : la segmentation en *chunks* et les étapes suivantes ne diffèrent que par les règles de composition utilisées. Le formalisme en triplet peut en effet être facilement utilisé pour la segmentation en *chunks* : il suffit d'étendre la notion de catégorie (au sens des grammaires catégorielles) aux deux premiers éléments du triplet.
2. Le second avantage est qu'il permet de mettre en parallèle des indices de différentes natures et de les associer tout au long du déroulement de l'analyse. Ces indices sont disponibles à chaque étape de l'analyse sans pour autant figurer dans la représentation sémantique finale (contrairement par exemple à la représentation rendue par un analyseur tel que TINA (cf. page 14)).

Dans LOGUS-II, ces indices contenus dans les éléments *C* et *R* sont respectivement de nature syntaxique et sémantico-pragmatique. Ils sont complémentaires et non réductibles l'un à l'autre : même si souvent, les objets correspondent à des groupes nominaux et les forces illocutoires à des groupes verbaux, il est impossible de faire de cette observation une

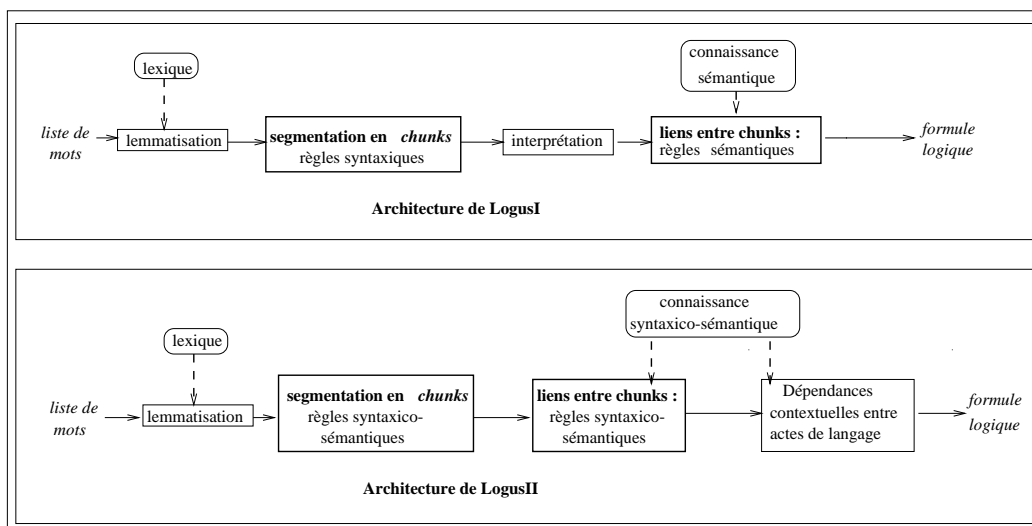


FIG. 6.5 – Architectures comparées de LOGUS-I et LOGUS-II

règle générale. Par exemple, dans l'énoncé « *comment on va au Louvre d'ici* », le *contenu propositionnel* est la chaîne d'objet

$$(\text{aller} [\text{vers} (\text{musee} [(\text{identite} (\text{nom} \text{ " Louvre" })])]), \\ (\text{venant_de} \text{ lieu_contexte})])$$

dont la catégorie syntaxique est un groupe verbal (*gv 3 sing*). La *force illocutoire* est indiquée par l'adverbe interrogatif « *comment* » (< *adverbe, interrogation, quomodo* >).

Par ailleurs, ce formalisme en triplet a été proposé pour tenir compte de la syntaxe et du rôle syntaxico-pragmatique ; un formalisme analogue pourrait permettre de prendre en compte des indices d'une autre nature : par exemple, on pourrait élargir le triplet en quadruplet pour intégrer un élément prosodique ; ce dernier pourrait être utilisé dans certaines règles pour aider à lever certaines ambiguïtés.

3. Enfin, ce formalisme permet de définir des règles syntaxico-sémantiques indépendantes du domaine de l'application, suivant les principes qui seront exposés dans le paragraphe 6.1.3.

6.1.3 Les étapes de l'analyse

La figure 6.5 permet de comparer les architectures de LOGUS-I et de LOGUS-II. Deux modifications essentielles différencient les deux systèmes :

1. La première est, comme il vient d'être dit, la disparition de l'étape d'interprétation qui suit la segmentation en *chunks* dans LOGUS-I.
2. Par contre, l'analyse faite dans LOGUS-II comporte une étape supplémentaire ; elle a pour objet l'étude des dépendances entre les différents actes de langage que peut contenir un énoncé. Ce traitement permet en particulier de tester une première mise en œuvre d'interprétation contextuelle (le contexte étant ici celui du dialogue) (§6.5, page 111).

6.2 La segmentation en *chunks*

6.2.1 Les rôles de la segmentation

La segmentation en *chunks* pratiquée dans LOGUS-II ne crée que des *chunks minimaux* (suivant le principe énoncé page 84). Elle joue néanmoins deux rôles essentiels :

1. Elle permet de rattacher à une tête lexicale les mots grammaticaux qui lui sont associés. La règle adoptée pour ces rattachements correspond aux observations des linguistes sur le *travail de formulation* (cf. page 52) : lorsque deux mots grammaticaux occupent la même place syntaxique, celui qu'il convient de prendre en compte est le dernier qui a été énoncé.
2. L'application de la règle précédente permet d'opérer un premier traitement des réparations lorsque le *reparandum* concerne des mots grammaticaux du type déterminants, prépositions, etc., normalement situés à proximité immédiate des têtes lexicales auxquelles ils sont liés : ces mots sont éliminés à l'issue de la segmentation lorsque celle-ci n'a pas permis leur rattachement (cf. §6.2.3, page 96).

6.2.2 Les règles

Les règles utilisées pour la segmentation en *chunks* sont complètement indépendantes du domaine de l'application. Par contre, les possibilités d'association entre les mots sont contenues dans les définitions du lexique.

Comme dans LOGUS-I, les règles utilisées sont directement inspirées des deux règles des grammaires catégorielles de type AB :

$$(A, A \setminus B) \rightarrow B \text{ et } (B/A, A) \rightarrow B$$

Dans LOGUS-II, ces règles sont généralisées aux deux premiers champs du triplet : la catégorie syntaxique et le rôle sémantique.

À partir des catégories syntaxiques et des rôles sémantiques simples présentés au §6.1.2, on définit des catégories et des rôles de type fractionnaires. La représentation sémantique d'un triplet dont la catégorie syntaxique et le rôle sémantique sont fractionnaires est une abstraction au sens des λ -termes (cf. annexe A). Par exemple, l'une des définitions du lexème « *le* » est $\langle (c_nom\ nomc\ (det\ def\ sing))/nomc, obj_acte/obj_acte, \lambda x.x \rangle$. D'une manière générale, ces catégories fractionnaires interviennent dans les définitions des mots grammaticaux.

Le regroupement des mots dans un *chunk* correspond alors à l'application des deux règles définies formellement de la façon suivante (F_C et F désignent des abstractions) :

$$\left. \begin{array}{l} \langle C_A, R_A, S_A \rangle, \\ \langle C_A \setminus F_C, R_A \setminus R_C, F \rangle \end{array} \right\} \rightarrow \langle (F_C\ C_A), R_C, (F\ S_A) \rangle$$

et

$$\left. \begin{array}{l} \langle F_C / C_A, R_B / R_A, F \rangle, \\ \langle C_A, R_A, S_A \rangle \end{array} \right\} \rightarrow \langle (F_C\ C_A), R_B, (F\ S_A) \rangle$$

Ces deux règles correspondent, en ce qui concerne les deux premiers champs du triplet aux règles des grammaires catégorielles précédentes. Pour ce qui est de la représentation sémantique

du triplet résultat, elle s'obtient par application de l'abstraction qui représente sémantiquement le triplet « fractionnaire » à la représentation sémantique du triplet atomique.

Dans l'exemple suivant, c_adj correspond à « *chunk adjectival* » et $prop$ signifie *propriete* (cf. figure 6.1 ou §6.1.2) :

$$\begin{array}{rcl}
 \langle \text{« pas »} & < (\lambda x.c_adj/adjectif), & (prop R)/(prop R), & (\lambda x.(pas x)) & \rangle \\
 \langle \text{« trop »} & < (\lambda x.x/adjectif), & (prop R)/(prop R), & \lambda x.x & \rangle \\
 \langle \text{« cher »} & < & adjectif, & (prop prix), & cher & \rangle \\
 \hline
 \langle \text{« pas trop cher »} & < & c_adj, & (prop prix), & (pas cher) & \rangle \\
 & & \text{avec } (\lambda x.c_adj (\lambda x.x adjectif)) \equiv_{\beta} c_adj \\
 & & \text{et } (\lambda x.(pas x) (\lambda x.x cher)) \equiv_{\beta} (pas cher)^2
 \end{array}$$

La définition de deux nouveaux types fractionnaires (notés respectivement \backslash_s et $/_s$) et celle de deux relations d'ordre partiel, la première sur les catégories syntaxiques (notée \leq_C) et la seconde (notée \leq_R) sur les rôles sémantiques, permettent la définition de deux nouvelles règles suivant les principes du calcul dans les pré-groupes initiés par Lambek (Buszkowski, 2001; Bargelli and Lambek, 2001).

$$\left. \begin{array}{l}
 \langle C_A, R_A, S_A \rangle, \\
 \langle C_B \backslash_s F_C, R_B \backslash_s R_C, F \rangle \\
 C_A \leq_C C_B, R_A \leq_R R_B
 \end{array} \right\} \rightarrow \langle (F_C C_A), R_C, (F S_A) \rangle$$

et

$$\left. \begin{array}{l}
 \langle F_C /_s C_C, R_B /_s R_C, F \rangle, \\
 \langle C_A, R_A, S_A \rangle \\
 C_A \leq_C C_C, R_A \leq_R R_C
 \end{array} \right\} \rightarrow \langle (F_C C_A), R_B, (F S_A) \rangle$$

L'intérêt essentiel de ces deux nouvelles règles est de permettre une diminution du nombre de définitions nécessaires des éléments de type fractionnaire pour couvrir l'ensemble des compositions possibles ; ces possibilités se déduisent des études sur les réparations et sur les erreurs de reconnaissance. Par exemple, l'ordre $nomc <_C (c_nom _ _) <_C (c_nom_p _ _)$ et la catégorie syntaxique $\lambda x.(c_nom_p x (preposition _))/_s(c_nom _ _)$ attribuée aux prépositions font que ces prépositions peuvent se composer avec un nom commun ou un *chunk* nominal non prépositionnel : les expressions « *vers Louvre* » (l'élision de mots courts lors de la reconnaissance est un phénomène fréquent), « *vers ce restaurant* » constituent des *chunks*. Par contre, dans une expression telle que « *au vers le Louvre* », la composition de « *au* » avec « *vers le Louvre* » est impossible : la préposition prise en compte est donc « *vers* ».

6.2.3 Mise en œuvre

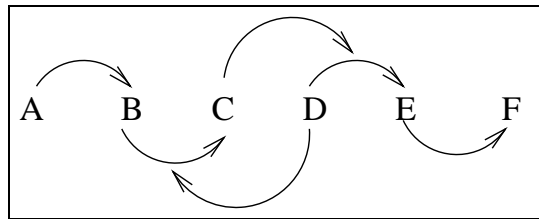
Les *chunks* sont obtenus par une analyse ascendante : la segmentation d'un énoncé correspond à la saturation des règles de compositions. La taille des *chunks* est donc déterminée par les définitions du lexique qui contiennent les compositions possibles.

2. Dans la mise en œuvre de LOGUS-II, la β -réduction est assurée par le langage de programmation λ Prolog qui est très brièvement présenté dans l'annexe B.

Application des règles

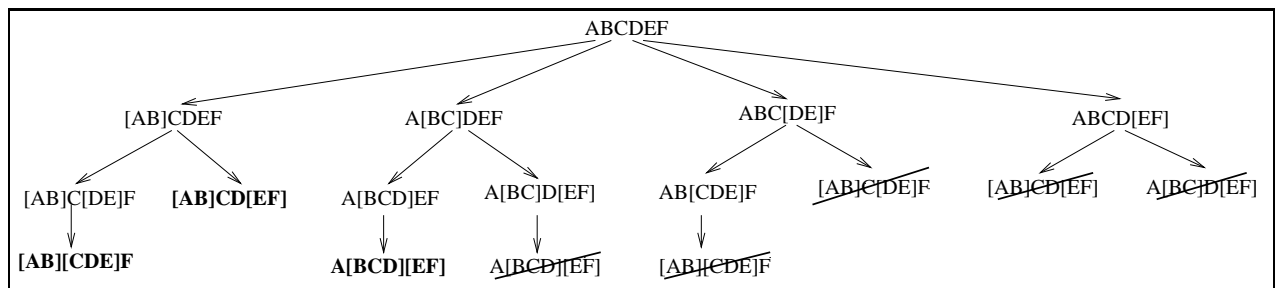
La mise en œuvre des règles de la segmentation consiste à appliquer toutes les compositions possibles jusqu'à épuisement de ces compositions. Les points fixes ainsi atteints sont appelés « solutions » de la segmentation en *chunks*. La recherche s'effectue en largeur dans l'arbre des compositions.

Considérons par exemple une liste (théorique) de 6 mots [ABCDEF], où les compositions possibles sont les suivantes :



$$\frac{A B}{[AB]}, \frac{B C}{[BC]}, \frac{D E}{[DE]}, \frac{E F}{[EF]}, \frac{[BC] D}{[BCD]}, \frac{C [DE]}{[CDE]}$$

Trouver les solutions de la segmentation correspond à un parcours d'arbre :



Une exploration de l'arbre en largeur d'abord permet, à chaque étape, d'éliminer les branches qui correspondent à des solutions répétées (rayées sur l'arbre ci-dessus). Les solutions de la segmentation en *chunks* sont alors données par les feuilles de l'arbre. Dans l'exemple précédent, l'application des règles correspond ainsi à trois solutions qui sont $[AB][CDE]F$, $[AB]CD[EF]$ et $A[BCD][EF]$.

Malgré le parcours en largeur qui permet les simplifications progressives, des explosions combinatoires sont possibles lorsque les énoncés sont assez longs (expérimentalement, lorsque leur longueur dépasse une quinzaine de mots). La solution adoptée est celle d'un découpage de l'énoncé en sous-énoncés ; les coupes sont introduites entre deux mots reconnus comme ne pouvant pas appartenir à un même *chunk*, par exemple, devant certaines prépositions ou marques de reprises (la recherche d'un tel découpage est assez facile car scinder l'énoncé en deux ou trois sous-énoncés est en général suffisant). Il est évident que les solutions de la segmentation en *chunks* s'obtiennent par juxtaposition des solutions des sous-énoncés.

Filtrage - Solutions *optimales*

- Les constituants des solutions de la segmentation subissent un *filtrage*. Celui-ci consiste en l'élimination d'un certain nombre de mots grammaticaux (par exemple les déterminants non utilisés) et, en particulier, de tous les constituants comportant des catégories et des rôles fractionnaires.

Ce traitement est essentiellement destiné à opérer un premier traitement des réparations (annoncé au §6.2.1) en éliminant des amorces lexicalement vides dues à la recherche des mots. Par exemple la segmentation appliquée à l'expression « *dans ce dans cet hôtel* » conduit à la construction de trois *chunks* : [*dans*] [*ce*] [*dans cet hôtel*]. Les deux premiers de ces *chunks* sont lexicalement vides et ils disparaissent à l'issue de l'opération de *filtrage*.

Ce procédé conduit à un « nettoyage » des productions orales qui présente deux particularités par rapport aux traitements de ce type qui ont été évoqués (cf. page 35) :

1. Il intervient après une première analyse (essentiellement syntaxique) de l'énoncé.
2. Il n'élimine aucun constituant lexical.

Ces deux différences en font un « nettoyage » non destructeur et pratiquement sans risque.

- Deux critères sont appliqués pour sélectionner les solutions de la segmentation ainsi obtenues. Ainsi une solution est dite *optimale* si et seulement si elle répond aux deux conditions suivantes :
 1. comporter un nombre minimum de constituants avant l'opération de filtrage,
 2. comporter un nombre maximum de constituants après le filtrage.

Ainsi, dans l'exemple, la solution [*AB*]*CD*[*EF*] n'est pas optimale puisqu'elle comporte 4 constituants alors que les autres solutions n'en comportent que 3. Le premier critère correspond à l'heuristique selon laquelle une solution est meilleure si elle correspond à un nombre maximum de compositions possibles. Le second critère correspond à l'heuristique selon laquelle le plus grand nombre possible de mots grammaticaux de l'énoncé doit être pris en compte.

Ambiguïtés

Chaque lexème peut posséder deux ou plusieurs définitions. La segmentation en *chunks* préserve des ambiguïtés qui peuvent se manifester sous deux formes différentes :

- Il peut arriver que deux ou plusieurs compositions soient possibles entre deux composants donnés : les résultats de ces compositions sont conservées en parallèle (un composant correspond à une liste de triplets qui représentent chacun une définition possible).
- Par ailleurs, toutes les solutions optimales de la segmentation en *chunks* au sens défini précédemment sont conservées.

Dans la pratique, on observe que les ambiguïtés liées à la première situation sont extrêmement fréquentes. À l'inverse, le caractère minimaliste des *chunks* fait que l'on observe rarement plus d'une solution optimale. Les tableaux suivants donnent quelques indications sur la fréquence de ces ambiguïtés dans un échantillon de 100 énoncés des séries 3 et 4 de l'évaluation « par défi »

(cf. page 116) au vocabulaire assez varié.

Nombre d'énoncés	Nombre total de <i>chunks</i>	Nombre de <i>chunks</i> ambigus
100	690	48

Nombre de définitions par <i>chunk</i>	1	2	3	≥ 3
Nombre de <i>chunks</i>	642	42	6	0

Nombre de <i>chunks</i> ambigus par énoncé	0	1	2	3	4
Nombre d'énoncés	59	36	4	0	1

Si 41 des 100 énoncés de l'échantillon présentent une ambiguïté liée à la définition des *chunks*, aucun d'entre eux n'admet plus d'une segmentation.

Donc, en général, il existe une seule segmentation de l'énoncé où chaque segment correspond à une, deux ou plusieurs définitions.

6.2.4 Exemple

Le tableau de la figure 6.6 donne un exemple de la segmentation en *chunks* sur un énoncé complet. Pour simplifier le tableau, seule la définition qui sert à l'élaboration de la solution optimale est représentée, en face du lexème concerné. Ce tableau indique le découpage de l'énoncé en *chunks*; le triplet qui correspond à chacun des *chunks* est écrit en caractères gras. Les compositions font apparaître 15 *chunks minimaux*. Le premier d'entre eux disparaît par filtrage à l'issue de la segmentation et les 14 autres sont soumis aux règles syntaxico-sémantiques pour la suite de l'analyse (donnée dans les tableaux des figures 6.8, 6.10, 6.11 et 6.13).

6.3 La connaissance syntaxico-sémantique

Les étapes de l'analyse qui suivent la segmentation en *chunks* utilisent une connaissance sémantico-syntaxique des dépendances qui peuvent exister entre les objets du domaine et leurs propriétés.

Une partie de cette connaissance peut être considérée comme une ontologie : les dépendances, purement sémantiques, correspondent essentiellement à des appartenances ou à des inclusions. Une autre partie permet de définir des dépendances moins sémantiquement évidentes, nécessairement marquées dans l'énoncé par des indices syntaxiques.

6.3.1 Dépendances sémantiques

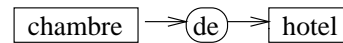
L'essentiel de la connaissance sémantique sur les objets du domaine est contenue dans la définition des deux prédicats suivants :

« je »	< (pronom pers 1 sing), annexe, neutre >
	∅
« j ai » « réservé »	< $\lambda x.(gv\ 1\ passe)/(pp\ "avoir")$, < info,obj_acte > /obj_acte, $\lambda x.(presence\ x)$ > < (pp "avoir"), obj_acte, (reservation []) >
	< (gv 1 passe), < info, obj_acte > , (presence (reservation [])) >
« une » « chambre »	< $\lambda x.(c_nom\ x\ (det\ indef\ sing))/nomc, obj_acte/obj_acte, \lambda x.x$ > < nomc, obj_acte, (chambre []) >
	< (c_nom nomc (det indef sing)), obj_acte, (chambre []) >
« dans » « un » « deux »	< $\lambda x.(pre_c_nom_p\ x\ (prep.\ in))/(pre_c_nom_), \lambda x.x/_ , \lambda x.x$ > < $\lambda x.(pre_c_nom\ (det\ indef\ sing))/(adj_num_), (prop\ A)/(prop\ A), \lambda x.x/_$ > < (adj_num inf_10), (prop int), (entier 2) >
	< (pre_c_nom_p (pre_c_nom (adj_num infl0) (det indef sing)) (prep. in)), (prop int), (entier 2) >
« étoiles »	< nomc, (unite objet), $\lambda x\ (hotel\ [(nbetoiles\ x)])$ >
	< nomc, (unite obj_acte), $\lambda x\ (hotel\ [(nbetoiles\ x)])$ >
« à » « l » « hôtel »	< $\lambda x.(c_nom_p\ x\ (prep.\ a_prep))/(c_nom_), R/R, \lambda x.x$ > < $\lambda x.(c_nom\ x\ (det\ def\ sing))/nomc, obj_acte/obj_acte, \lambda x.x$ > < nomc, obj_acte, (hotel []) >
	< (c_nom_p (c_nom nomc (det def sing)) (prep. a_prep)), obj_acte, (hotel, []) >
« euh »	< particule, hesitation, neutre >
	< particule, hesitation, neutre >
« à » « l » « hôtel »	< $\lambda x.(c_nom_p\ x\ (prep.\ a_prep))/(c_nom_), R/R, \lambda x.x$ > < $\lambda x.(c_nom\ x\ (det\ def\ sing))/nomc, obj_acte/obj_acte, \lambda x.x$ > < nomc, obj_acte, (hotel []) >
	< (c_nom_p (c_nom nomc (det def sing)) (prep. a_prep)), obj_acte, (hotel []) >
« Caumartin »	< nomp, (prop identite), (nom "Caumartin") >
	< (nomp, (prop identite), (nom "Caumartin")) >
« pour » « y aller »	< $\lambda x.(infinitif_p\ (prep.\ pour))/infinitif, obj_acte/obj_acte, \lambda x.x$ > < infinitif, obj_acte, (aller [(vers obj_contexte)]) > ,
	< (infinitif_p (prep. pour)), obj_acte, (aller [vers obj_contexte]) >
« d » « ici »	< $\lambda x.(c_nom_p\ x\ (prep.\ ab))/(pronom\ lieu\ 3\ sing),obj_acte/obj_acte,\lambda x.x$ > < (pronom lieu 3 sing), obj_acte, lieu_contexte >
	< (c_nom_p (pronom lieu 3 sing) (prep. ab)), obj_acte, lieu_contexte >
« comment »	< adverbe, interrogation, comment >
	< adverbe, interrogation, comment >
« est-ce-que »	< expr, interrogation, oui_non >
	< expr, interrogation, oui_non >
« je » « peux »	< (gv 1 B)/(verbe _ B), R/R, $\lambda x.x$ > < (verb 1 pres.), modalite, pouvoir >
	< (gv 1 pres.), modalite, pouvoir >
« faire »	< infinitif, annexe, neutre >
	< infinitif, annexe, neutre >

FIG. 6.6 – Exemple de segmentation en chunks sur un énoncé complet

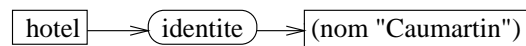
- Le prédicat *est_sous_obj_de* exprime l'existence d'une relation d'appartenance ou d'inclusion entre deux objets. Il s'agit d'un prédicat d'arité 2 qui prend en argument deux étiquettes d'objet ; il est donc défini sur $\mathcal{E}t_Obj \times \mathcal{E}t_Obj$ (cf. page 65).

La relation $est_sous_obj_de(Et_1, Et_2)$ signifie qu'un objet simple O_1 d'étiquette Et_1 et un objet O_2 d'étiquette Et_2 peuvent être liés par la relation d'enchaînement *de* pour former la chaîne d'objets (*de* O_1 O_2). Chaque instantiation du prédicat correspond à la possibilité de construire un graphe conceptuel élémentaire ; par exemple, $est_sous_obj_de(chambre, hotel)$ correspond à :



- Le prédicat *est_propriete_de* est un prédicat d'arité 3 défini sur $\mathcal{E}t_Prop \times \mathcal{S}em_Prop \times \mathcal{E}t_Obj$. Ses deux premiers arguments sont donc une étiquette de propriété et sa détermination ; le troisième est une étiquette d'objet (cf. page 65).

La relation $est_propriete_de(EP, P, Et)$ signifie que la propriété d'étiquette EP et de détermination P peut être une propriété d'un objet O d'étiquette Et . La propriété (EP, P) peut donc être ajoutée à la liste des propriétés de l'objet O . Chaque instantiation correspond à la construction du graphe conceptuel élémentaire. Par exemple, $(est_propriete_de, identite, (nom "Caumartin"))$ permet de construire le graphe conceptuel élémentaire :



6.3.2 Dépendances syntaxico-sémantiques

Les dépendances entre objets ne se limitent pas à des relations d'appartenance ou d'inclusion ; d'autres relations de dépendances existent, modélisées dans la langue cible comme des propriétés particulières (cf. §5.2.2). En général, la détermination de ces dépendances fait appel à des critères syntaxiques. Par exemple, les deux énoncés suivants contiennent le même syntagme nominal prépositionnel « à l'hôtel Caumartin » :

1. « À l'hôtel Caumartin les tarifs c'est quoi »
2. « Quel métro je peux prendre pour aller à l'hôtel Caumartin »

La relation $est_sous_obj_de(tarif, hotel)$ permet de construire la chaîne :

$(de (tarif, []) (hotel [(identite (nom "Caumartin"))]))$

Dans le deuxième énoncé « à l'hôtel Caumartin » peut être considérée comme une propriété de direction de l'objet (acte) d'étiquette *aller*, ce qui correspond à la construction de l'objet :

$(aller [(vers (hotel [(identite (nom "Caumartin"))]))]))$

Cette interprétation du syntagme est liée à la présence de la préposition *à* : dans LOGUS-II, cette préposition est un attribut de la *catégorie syntaxique* du constituant correspondant à ce syntagme.

Pour tenir compte de ce besoin, le prédicat *est_dependant_de* est un prédicat d'arité 4 : *est_dependant_de*(*C1*, *O1*, *Et*, *Lien*) correspond à la possibilité que l'objet *O1* de catégorie syntaxique *C1* puisse être le complément (donc traité comme une propriété particulière) d'un objet *O* d'étiquette *Et*, la relation d'objets à propriétés étant *Lien* (*Lien* joue donc le rôle d'une étiquette de propriété). Le prédicat est donc défini sur :

$$Categ \times Et_Obj \times Objet \times Rel_obj_Prop.$$

Par exemple, *est_dependant_de*(*Cat_CO*, (*EtOLPO*), *aller*, *vers*) si

1. *Cat_CO* contient la préposition *ad*,
2. *LPO* contient une propriété d'étiquette *identite*.

La relation correspond à un graphe conceptuel de la forme suivante :



Le prédicat *est_dependant_de* peut parfois servir à définir une relation de dépendance directe. Par exemple, la juxtaposition des deux concepts *liste* et *hotel* renvoie préférentiellement à la chaîne d'objets (*de* (*liste* []) (*hotel* [])). Cependant, l'expression « *les hôtels de la liste...* » renvoie à une chaîne du type (*de* (*hotel* []) (*liste* [])). La forme de la dépendance est syntaxiquement marquée par l'ordre des constituants mais surtout par le déterminant et la préposition (on pourrait avoir aussi « *dans cette liste les hôtels* »). Le prédicat *est_dependant_de* sert aussi à rendre compte de l'éventualité de ce type de dépendance.

6.3.3 Annexes et conséquences

Des relations peuvent être définies comme des conséquences de celles établies par les prédicats de la connaissance sémantique. Par exemple, la gestion des coordinations et des « réparations » implique la détermination de leur portée (la similitude entre ces deux familles de phénomènes a été évoquée dans le chapitre 4, page 53). Les indices syntaxiques tels que les déterminants ou les prépositions ne sont pas toujours suffisants et de plus ils sont souvent « bruités » par les erreurs de reconnaissance. Un indice supplémentaire important est la sémantique. Les éléments réparés et l'élément qu'ils réparent appartiennent en général à une même famille syntaxique et sémantique ; en général, il en est de même des éléments coordonnés.

Les prédicats suivants ont pour but d'indiquer si deux objets ou deux propriétés peuvent être sémantiquement coordonnés ou se réparer l'un l'autre. Le premier prédicat correspond à un indice sémantique fort de coordination ou de réparation. Le second est plutôt une non-impossibilité sémantique.

meme_famille_sem :

- La relation est vérifiée pour deux propriétés de même étiquette.
- La relation est vérifiée entre deux objets si
 1. ils ont la même étiquette ou si

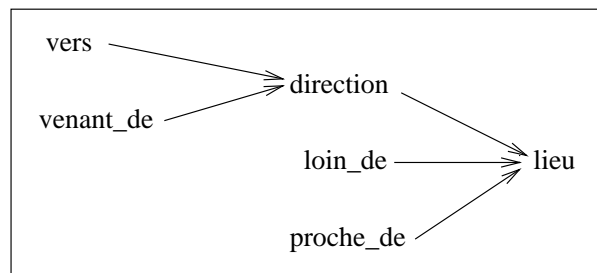


FIG. 6.7 – Relation d'ordre partiel sur les étiquettes des propriétés

2. ils ne sont pas subordonnables l'un à l'autre (ils ne peuvent pas être deux éléments distincts d'une même chaîne simple) ou si
3. ils peuvent posséder un même sous-objet commun ou ils peuvent être sous-objets d'un même objet.

compatible_famille_sem :

- La relation est vérifiée pour deux propriétés qui peuvent être rattachées à un même objet.
- La relation est vérifiée entre deux objets si
 1. ils sont non identiques ou si
 2. ils ne sont pas subordonnables l'un à l'autre (ils ne peuvent pas être deux éléments distincts d'une même chaîne simple).

La connaissance sémantique liée aux objets et à leurs propriétés comprend également quelques prédicats annexes qui permettent des comparaisons ou des simplifications. Par exemple, les étiquettes des propriétés sont munies d'une relation d'ordre partiel ; celui-ci permet ainsi de savoir que l'étiquette *lieu* est un élément maximal de l'ensemble $\{vers, venant_de, direction, loin_de, proche_de, lieu\}$ suivant le schéma de la figure 6.7. Ce prédicat joue un rôle analogue à celui joué par les prédicats *meme_famille_sem* et *compatible_famille_sem*. Il permet de déterminer si deux propriétés appartiennent ou non à la même famille sémantique ; il fournit ainsi un indice sémantique pour leur traitement dans une coordination ou une réparation. L'ordre correspond à la précision de l'étiquette ; il peut intervenir dans la définition de l'élément résultant obtenu (par exemple, la règle donnée page 104 fait intervenir la borne sup. des étiquettes des deux propriétés).

6.4 Les règles syntaxico-sémantiques

6.4.1 Définition des règles

À l'issue de la segmentation en *chunks*, les constituants obtenus contiennent au plus un *mot lexical* et ils peuvent être classés en trois grandes catégories :

1. Les *chunks* qui contiennent un élément lexical au sens utilisé dans la règle qui définit les *chunks minimaux* (page 84), à savoir une tête lexicale à laquelle sont éventuellement rattachés des mots grammaticaux (prépositions, déterminants, pronoms, etc.), tels que :

[« dans le restaurant »], [« vers le Louvre »], [« le plus proche »], [« dans un trois »], [« moins de vingt »], [« pas trop cher »], [« je voudrais »].

2. Les *chunks* que l'on peut qualifier de *grammaticaux* : pronoms relatifs, marques de coordination, d'hésitation ou de reprise, négations (lorsqu'elles n'ont pas été rattachées à une tête lexicale), etc.
3. Les mots inconnus du vocabulaire de la compréhension ou considérés comme *neutre* (cf. page 90), c'est à dire sans correspondance sémantique dans le langage cible qui a été défini.

La connaissance syntaxico-sémantique constitue une base de données des liens qui peuvent être établis entre les objets du domaine ou entre les objets et leurs propriétés pour la construction de la formule finale. Les règles permettent la mise en œuvre de ces liens. Comme elles portent sur plusieurs éléments consécutifs de l'énoncé, elles prennent en compte l'ordre de ses différents constituants. Par ailleurs, c'est également par l'intermédiaire des règles que sont gérées les marques de structure de cet énoncé telles que coordinations, marques de reprise, d'hésitations ou d'autocorrections, pronoms relatifs, etc.

Niveaux de règles

Les études sur la langue parlée montrent le respect de structures syntaxiques locales et cette propriété a été utilisée pour la segmentation en *chunks*. Elles indiquent aussi qu'il faut s'attendre à certaines « particularités » dans les énoncés de l'oral spontané : télescopages syntaxiques, mouvements de constituants, changements soudains de parcours syntaxiques ou sémantiques, énoncés parenthétiques (incises) (Kerbrat-Orecchioni, 1990; Blanche-Benveniste, 1990). Cependant, elles attestent qu'il est « inutile d'envisager une syntaxe spécifique de l'oral » (Blanche-Benveniste, 2002).

Les « particularités » précédentes, auxquelles viennent s'ajouter les erreurs de reconnaissance, font que les règles syntaxico-sémantiques utilisées pour composer les *chunks* doivent être tolérantes et flexibles. Mais en même temps, il semble aller de soi qu'il faille privilégier les compositions pour lesquelles les indices syntaxiques « classiques » (ordre des constituants, usage des prépositions, etc.) et sémantiques coïncident.

Pour mettre en œuvre ce principe de règles préférentielles, la solution retenue dans LOGUS-II est celle de mettre en œuvre une analyse incrémentale ; celle-ci correspond à la définition de priorités dans les compositions par des « niveaux » de règles.

Par exemple, l'un des problèmes les plus délicats que doit résoudre l'analyse est de pouvoir identifier la portée des réparations et des coordinations. Le principe appliqué est qu'en général on peut observer une certaine homogénéité entre les éléments repris, corrigés ou coordonnés ; celle-ci peut se manifester syntaxiquement (par exemple, reprises des prépositions) et sémantiquement. Les règles correspondant aux coordinations et réparations sont définies à différents niveaux, en fonction de la convergence des indices syntaxiques et sémantiques comme on peut le voir dans les exemples qui suivent.

Forme générale

Une règle permet de transformer une liste de n ($2 \leq n \leq 4$) constituants consécutifs de l'énoncé $[E_1, E_2, \dots, E_n]$ où $E_i = \langle C_i, S_i, T_i \rangle$ en une liste de m constituants $[F_1, \dots, F_m]$ avec $m < n$. Pour cela, les E_i doivent vérifier un ensemble de conditions $Cond = \{cond_1, \dots, cond_k\}$, où $k \in \mathbb{N}$. Ces conditions ne doivent porter que sur les C_i et T_i , $i \in \{1, \dots, n\}$ ou correspondre à une valeur de vérité d'un prédicat de la connaissance sémantique appliqué aux C_i et T_i . Les constituants résultats F_1, \dots, F_m sont obtenus à partir d'un ensemble de règles de constructions $Constr = \{constr_1, \dots, constr_p\}$ avec $p \in \mathbb{N}$ appliquées aux constituants E_1, E_2, \dots, E_n . Il convient de noter que la définition de ces règles ne fait pas directement intervenir les traductions sémantiques T_i des E_i .

Par la suite, les règles sont représentées sous la forme suivante :

$$\frac{[E_1, \dots, E_n]}{Cond} \frac{[F_1, \dots, F_m]}{Constr}$$

6.4.2 Les différents niveaux

Premier niveau

Le premier niveau permet essentiellement la construction de *chunks* classiques à partir des *chunks minimaux*. Il ne contient aucune règle permettant de fabriquer une chaîne d'objets composée et les seules propriétés qui peuvent être rattachées à un objet sont celles qui sont contenues dans des adjectifs précédents le nom auquel ils se rattachent (« *pour deux chambres* », « *dans un bon hôtel* » ...).

Les règles permettent de réaliser les associations suivantes :

- Chunks lexicaux juxtaposés occupant la même position sémantique :
 - * la liaison peut être indiquée par une ou des marques de coordination, de reprise ou de négation qui est située avant, entre ou après les deux *chunks* lexicaux :
 - [« *avec douche* »] [« *ou* »] [« *salle_de_bain* »],
 - [« *deux* »] [« *non* »] [« *trois* »],
 - [« *deux* »] [« *non* »] [« *trois* »] [« *plutôt* »],
 - [« *pas* »] [« *le Louvre* »] [« *le Grévin* »]
 - * énumérations ; dans les deux exemples suivants, deux compositions sont appliquées successivement :
 - [« *deux* »] [« *trois* »] [« *ou* »] [« *quatre* »]
 - [« *deux* »] [« *trois ou quatre* »],
 - [« *le bus* »] [« *le métro* »] [« *et* »] [« *le RER* »]
 - [« *le bus* »] [« *le métro et le RER* »]
- Chunks lexicaux juxtaposés « *complémentaires* » :
 - * le premier *chunk* est le début du groupe nominal dont la tête lexicale est dans le second (en général, il s'agit d'un groupe adjectival qui qualifie un groupe nominal) :

[« moins de deux cents »] [« euros »]
 [« dans un deux »] [« étoiles »]
 [« un grand »] [« restaurant »]
 [« le plus proche »] [« des grands magasins »]

* le second *chunk* est un complément du premier :

[« pas trop loin »] [« de la gare »]
 [« le plus proche »] [« des grands magasins »]
 [« moins cher »] [« que »] [« le Caumartin »]

– Simplifications :

deux ou plusieurs *chunks* successifs servent à désigner une coordination ou une réparation :

[« plutôt »] [« non »] [« pas »]
 [« non »] [« je veux dire »]

Par exemple, la coordination entre deux propriétés de même rôle ou de rôles comparables (« deux ou trois », « avec douche ou salle de bain ») est réalisée avec la règle suivante³ :

$$\frac{[< C_1, (prop EP_1), T_1 >, < _, (coordination Co), neutre >, < C_2, (prop EP_2), T_2 >] \quad Cond = \{(EP_1 = EP_2 \text{ ou } EP_1 \text{ et } EP_2 \text{ comparables})\}}{[< conj C_1 C_2 Co >, (prop EP), T >] \quad Constr = \left\{ \begin{array}{l} \text{si } T_1 = T_2 \text{ alors } T = T_1 \text{ sinon } T = (Co T_1 T_2), \\ EP = \text{borne sup de } EP_1 \text{ et } EP_2 \end{array} \right\}}$$

Second niveau

Si les constituants obtenus à l'issue du premier niveau correspondent à peu près à des *chunks* classiques, les règles du second niveau permettent de réaliser toutes les compositions syntaxiquement et sémantiquement correctes entre ces *chunks*. Elles permettent la construction des chaînes d'objets et la coordination de ces chaînes, à condition toutefois que les indices syntaxiques et sémantiques convergent. Elles doivent donc permettre, en principe, de relier entre eux tous les syntagmes d'un énoncé lorsque celui-ci respecte certaines conditions de « correction » imposées.

Elles peuvent être classées en deux grandes catégories suivant la présence ou non de marqueurs extérieurs aux *chunks* : coordinations, marques d'hésitation ou de reprise, etc, permettant de préciser la nature de la liaison. Ainsi, la plupart d'entre elles prennent en entrée une liste d'éléments de l'une des formes suivantes :

1. Simple juxtaposition

(a) objet/objet

Exemples : [« un ticket »] [« de métro »],
 [« visiter »] [« le Louvre »],
 [« la ligne »] [« de bus ou de RER »]

3. Les notions de *comparables* et de *borne sup* sont liées à la relation d'ordre sur les étiquettes de propriétés définie page 101.

- (b) objet/propriete ou propriete/objet
Exemples : [« un hôtel »] [« deux étoiles »],
[« une pharmacie »] [« encore ouverte »]
- (c) indice de l'intention du locuteur (interrogation, requête, refus etc.), objet ou propriete
Exemples : [« quels sont »] [« les cinémas »]
[« y a-t-il »] [« des restaurants »]
- (d) propriete/propriete
Exemples : [« deux »] [« avec douche »]
- (e) indice de l'intention du locuteur/indice de l'intention du locuteur
Exemples : [« est-ce que »] [« je peux »]
[« vous pouvez »] [« me dire si »]

2. Avec marque de coordination (au sens large)

- (a) objet/marque de coordination/objet
Exemples : [« un musée »] [« ou »] [« un cinéma »]
[« un musée »] [« non »] [« un cinéma »]
- (b) marque de coordination/objet/objet
Exemples : [« pas »] [« un musée »] [« un cinéma »]
- (c) objet/objet/marque de coordination
Exemples : [« un musée »] [« un cinéma »] [« plutôt »]
- (d) propriété/marque de coordination/propriété
Exemples : [« pas trop loin »] [« et »] [« pas trop cher »]
[« près de la gare »] [« je veux dire »]
[« à côté de la gare Montparnasse »]
- (e) marque de coordination/propriété/propriété
Exemples : [« pas »] [« près du Louvre »] [« à côté de la gare »]
- (f) propriété/propriété/marque de coordination
Exemples : [« pas trop cher »] [« une ou deux étoiles »] [« je veux dire »]

Par exemple, l'analyse de l'expression « l'adresse d'un deux étoiles et ses tarifs » correspond à l'application successive des deux règles suivantes⁴

– Règle 1 :

$$\begin{array}{l}
 [\langle C1, \text{obj_acte}, O1 \rangle , \langle C2, \text{obj_acte}, O2 \rangle] \\
 \text{Cond} = \left\{ \begin{array}{l}
 -O1 \text{ objet simple d'étiquette } Et1 \\
 -O2 \text{ chaîne simple d'étiquette } Et2 \\
 -\text{est_sous_obj_de}(Et1, Et2)
 \end{array} \right. \\
 \hline
 [(\text{categs } C1 \ C2), \text{obj_acte}, (\text{de } O1 \ O2)]
 \end{array}$$

4. Les règles sont volontairement assez précises, chacune correspondant à un traitement précis. En effet, LOGUS fournit, avec les résultats de l'analyse, la liste des règles utilisées. Définir un plus grand nombre de règles rend ainsi plus simple l'identification des traitements effectués.

– Règle 2 :

$$\begin{array}{l}
 \left[\langle C1, \text{obj_acte}, O1 \rangle, \langle _, (\text{coordination } \textit{et}), _ \rangle, \langle C2, \text{obj_acte}, O2 \rangle \right] \\
 \text{Cond} = \begin{cases} -C1 \text{ et } C2 \text{ sans préposition ou avec la même préposition} \\ - \textit{meme_famille_sem}(O1, O2) \end{cases} \\
 \hline
 \left[\langle \textit{conj } C1 C2 \textit{ et} \rangle, \text{obj_acte}, (\textit{et } RO1 RO2) \right] \\
 \text{Cond} = [RO1 \text{ et } RO2 \text{ objets obtenus par } \textit{factorisation}]
 \end{array}$$

- La première règle a pour conséquence la fabrication de la chaîne d’objets

(de (adresse []) (hotel (nbetoiles (entier 2))))

- La deuxième règle permet de coordonner la chaîne précédente avec l’objet simple qui correspond à « *ses tarifs* ». Les indices syntaxiques et sémantiques coïncident et indiquent que la coordination s’applique à *adresse/tarif*. L’objet (*et RO1 RO2*) s’obtient par *factorisation* suivant le principe énoncé page 68.

Troisième niveau

Les règles du troisième niveau permettent d’opérer des compositions similaires à celles du second niveau, tout en augmentant la flexibilité :

- Elles offrent la possibilité de composer des constituants en « sautant » certains éléments jugés non essentiels, soit à cause de leur rôle sémantique : mots inconnus, marques d’hésitation ou de reprise, (*r_inconnu, excuse, hesitation, reprise, auxiliaire*), coordinations ((*coordination _*)), etc, soit à cause de leur catégorie syntaxique qui aurait dû conduire à leur composition avec des éléments situés à proximité immédiate au cours des niveaux précédents (*pre_c_nom, pre_c_nom_p*, etc).
« [je voudrais un restaurant pas trop chers] [sont] [adresse] ... » (erreur de reconnaissance *son/sont* obtenue avec *viavoice*, cf. page 121)
- Tolérance dans l’ordre des constituants :
« [avec douche] [une chambre] »
« [le Louvre] [quelles sont ses heures d’ouverture] »
- Tolérance dans l’utilisation des déterminants et des prépositions :
« [les tarifs dans un deux étoiles] [et] [un trois étoiles aussi] »
Dans ce dernier exemple la coordination entre les constituants « *dans un deux étoiles* » et « *un trois étoiles* » n’a pas été opérée au second niveau à cause de la non-répétition de la préposition « *dans* ».

Le troisième niveau donne aussi une mesure de la flexibilité requise pour produire chaque interprétation qui est utilisée pour choisir la ou les solutions retenues.

Cascades de règles

La définition des différents niveaux permet une application des règles « en cascades », qui correspond à une analyse incrémentale de l'énoncé. Chacun des niveaux peut produire plusieurs solutions, qui sont filtrées selon les principes appliqués lors de la segmentation en *chunks*. Toutes les compositions autorisées par les règles d'un niveau sont appliquées jusqu'à saturation et seules sont retenues pour le niveau suivant les *solutions optimales*, à savoir celles qui produisent un nombre minimal de constituants. De plus, le troisième niveau ne retient que les solutions optimales dont la mesure de flexibilité est la plus basse.

6.4.3 Exemple

Le tableau de la figure 6.8 indique les constituants résultants de l'énoncé déjà utilisé comme exemple pour la segmentation en *chunks* (cf. figure 6.6, page 98) après application des règles du niveau 1. Un schéma des règles qui ont été utilisées est donnée dans le tableau de la figure 6.9. Ce niveau permet la reconstitution des syntagmes nominaux complets « *dans un deux étoiles* » et « *à l'hôtel euh à l'hôtel Caumartin* » : on obtient donc la segmentation obtenue dans une segmentation en *chunks* « classique ».

Le tableau de la figure 6.10 donne le résultat de l'application des règles de niveau 2. Pour construire la première partie de l'énoncé, la règle appliquée correspond à la règle 1 donnée page 105 ; elle suffit à l'analyse de la première partie de l'énoncé : « *je j'ai réservé une chambre dans un deux étoiles* ». Par contre, la non-répétition de la préposition « *dans* » devant l'expression « *à l'hôtel* » ont fait que les deux objets correspondants n'ont pas été composés. De même, la position de l'expression « *pour y aller d'ici* » n'a pas permis sa liaison avec « *comment est-ce que je peux faire* ».

Au niveau 3, deux règles sont appliquées : la première est un assouplissement de la règle qui permet de combiner deux objets juxtaposés (en l'occurrence, l'objet est répété avec un enrichissement lexical), malgré une modification de la préposition : elle permet de combiner « *dans un deux étoiles* » avec « *à l'hôtel Caumartin* » ; à ce stade de l'analyse, le premier de ces deux éléments est inséré dans une chaîne d'objets mais sa catégorie syntaxique qui contient la nature de la préposition est néanmoins disponible (cf. page 85). La seconde est un assouplissement dans l'ordre attendu des constituants. Après l'application de ces deux règles, l'énoncé est composé de deux constituants dont le rôle sémantique et la traduction sémantique figurent dans le tableau de la figure 6.11.

6.4.4 Noyaux sémantiques

L'étape de segmentation en *chunks* établit des dépendances essentiellement syntaxiques et de proximité immédiate. L'étape des règles syntaxico-sémantiques établit des liens entre *chunks* juxtaposés ou séparés par des éléments identifiés comme peu signifiants. Ces deux étapes suffisent à l'analyse des énoncés simples et pas trop disloqués ; à leur terme, tous les éléments de l'énoncé sont composés en un seul constituant.

L'étape suivante concerne donc les énoncés qui, à l'issue des deux traitements précédents, se présentent sous la forme de plusieurs constituants juxtaposés. Elle consiste à essayer de détecter et de délimiter les différents actes qui composent éventuellement cet énoncé.

« je j ai réservé » < (gv 1 passe), (roles info. obj_acte), < presence, (reservation, []) >>
« une chambre » < (c_nom (det indef sing)), obj_acte, (chambre []) >
« dans un deux étoiles » < (c_nom_p (c_nom (det indef sing)) (prep. in)), obj_acte, (hotel [(nbetoiles (entier 2))]) >
« à l hôtel euh à l hôtel Caumartin » < (c_nomp_p (prep. a_prep)), obj_acte, (hotel[(identite (nom Caumartin))]) >
« pour y aller » < (infinitif_p (prep. for)),obj_acte,(aller []) >
« d ici » < (c_nom_p (pronom lieu)(prep. from)),obj_acte,lieu_contexte >
« comment » < adverbe,interrogation,quomodo >
« est-ce-que » < expr,interrogation,yes_no >
« je peux » < (gv 1 pres.),modalite,pouvoir >
« faire » < infinitif,annexe,neutre >

FIG. 6.8 – Résultat de l'application des règles de niveau 1

1	« dans un deux » « étoiles » [< C1, (prop entier), T1 > , < nomc, (unite R), F >] Cond = [C1 ∈ [(pre_c_nom _), (pre_c_nom_p _)]] [< CR, R, (F T1) >] Constr = [CR chunk nominal prépositionnel]
2	« à l'hôtel » « euh » « à l'hôtel » [< C, R, T > , < _, hesitation, _ > , < C, R, T >] Cond = [R ∈ [(prop _), obj_acte]] [C, R, T]
3	« à l'hôtel » « Caumartin » [< C1, obj_acte, O1 > , < C2, (prop identite), T2 >] Cond = { [est_propriete_de(identite, T2, Et1) (Et1 : étiquette de O1), pas de propriété identite de détermination ≠ T2 dans O1, C1 ∈ {nomc, (c_nom _), (c_nom_p _)}] [< C1,obj_acte, RO1 >] Constr = { [RO1 obtenu en ajoutant (identite T2) à la liste des propriétés de O1]

FIG. 6.9 – Schéma (simplifié) des règles utilisées au niveau 1

« je j ai réservé une chambre dans un deux étoiles »
<pre>(categs (gv 1 passe)) (categs (c_nom (det indef sing) (c_nom_p (c_nom (det indef sing)) (prep. in))))), obj_acte, ((info presence) (de (reservation []) (de (chambre []) (hotel [(nbetoiles (entier 2))] >)))) ></pre>
« à l hôtel euh à l hôtel Caumartin »
<pre>< (c_nomp_p (prep. a_prep)), obj_acte, (hotel [(identite (nom "Caumartin"))]) ></pre>
« pour y aller d ici »
<pre>< (infinitif_p (prep.for)), obj_acte, (aller,[(vers obj_contexte),(venant_de lieu_contexte)]) ></pre>
« comment est-ce-que je peux faire »
<pre>< (gv 1 pres),interrogation,quomodo ></pre>

FIG. 6.10 – Résultat de l'application des règles de niveau 2

« j ai réservé une chambre dans un deux étoiles »
« à l hôtel euh à l hôtel Caumartin »
<pre>< info,obj_acte > < presence,(de (reserv. []) (de (chambre []) (hotel [(nbetoiles (entier 2)), (identite (nom "Caumartin"))])))) ></pre>
« pour y aller d ici comment est-ce-que je peux faire »
<pre>< interrogation, obj_acte > < quomodo, (aller [(vers obj_contexte)) < venant_de lieu_contexte >] ></pre>

FIG. 6.11 – Résultat de l'application des règles de niveau 3

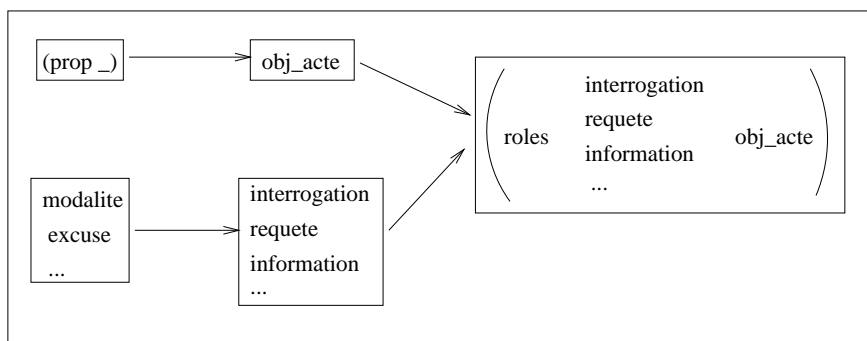


FIG. 6.12 – L'ordre sur les rôles sémantiques des noyaux

L'objectif à atteindre est double : il s'agit d'abord de remédier à la dislocation éventuelle de l'énoncé (due par exemple à une incise qui sépare deux éléments sémantiquement liés). Le traitement doit permettre de récupérer au moins une partie du sens de cet énoncé (et, autant que possible la partie essentielle de ce sens). Le second objectif est de permettre une analyse correcte des énoncés composés de plusieurs actes : une première partie du traitement consiste à les distinguer.

L'idée mise en œuvre est que le sens le plus probable de l'énoncé est celui qui correspond à une composition sémantiquement vraisemblable du plus grand nombre de ses éléments, la notion de « vraisemblance sémantique » étant déterminée par les liens définis dans la connaissance syntaxico-sémantique. Cette heuristique correspond à l'un des principes utilisés dans LOGUS-I et elle a semblé donner des résultats convenables.

En particulier, l'analyse cherche à construire les chaînes d'objets incluant le plus grand nombre possible des objets et des propriétés de l'énoncé (en respectant les contraintes définies par les prédicats de la connaissance syntaxico-sémantique). L'analyse peut n'être que partielle : le résultat correspond aux différentes associations opérées. Le traitement se déroule suivant les étapes successives suivantes :

1. Les règles des niveaux précédents ont généralement permis d'associer entre eux des éléments : l'énoncé peut ainsi comporter des marques de requêtes associées à des objets ou propriétés, des chaînes d'objets, etc. Ces éléments associés sont alors considérés comme des « *noyaux sémantiques* » de l'énoncé. Un ordre est défini sur les rôles sémantiques de ces noyaux : il est donné dans la figure 6.12 ; le plus élevé correspond à un acte de langage appliqué à une chaîne d'objets, les moins élevés est une simple propriété ou un indice d'un acte de langage.

Définition : Un noyau est dit **principal** s'il est de l'ordre le plus élevé possible dans l'énoncé. La première étape consiste à déterminer le premier *noyau principal* (le plus à gauche) de l'énoncé.

2. Ensuite, le traitement se poursuit par la recherche d'éventuels noyaux sémantiques *indépendants*. Cette notion d'*indépendance* est définie par rapport à des critères syntaxiques ou sémantiques et repose à la fois sur la catégorie syntaxique et le rôle sémantique des noyaux : deux groupes verbaux sont indépendants, deux noyaux de l'ordre le plus élevé sont indépendants, etc.

3. Enfin, l'analyse tente de rattacher sémantiquement les autres éléments de l'énoncé aux noyaux ainsi identifiés, en utilisant les prédicats de la connaissance syntaxico-sémantique (voir section 6.3). La seule contrainte imposée est une interdiction d'un croisement des rattachements entre deux noyaux (par exemple, si N_1 et N_2 sont deux noyaux indépendants, dans N_1ABN_2 , le rattachement de A à N_2 interdit celui de B à N_1).

Un élément isolé qui ne peut se rattacher à l'un des noyaux est simplement ignoré. Ainsi, ce traitement doit permettre d'extraire d'un énoncé apparemment disloqué des parties sémantiquement cohérentes, en prenant en compte le plus grand nombre possible de ses éléments.

Ce traitement opéré sur un énoncé disloqué peut faire penser à l'analyse sélective pratiquée dans TINA sur les énoncés pour lesquels une analyse plus classique a échoué (cf. page 14). Cependant, une différence essentielle est que, dans LOGUS, ce traitement n'est pas dissocié des traitements précédents : loin d'être un traitement alternatif utilisé en cas d'échec des étapes antérieures, il s'appuie fortement sur les résultats qui en sont issus. Il faut donc plutôt le concevoir comme une étape supplémentaire, où la contrainte de composer des éléments juxtaposés est relâchée. Cette particularité le différencie également de la stratégie adoptée dans les analyses sélectives où la recherche de la nature de l'acte de langage (à partir duquel l'analyse s'organise) constitue la première étape de l'analyse de l'énoncé.

6.5 Contexte et dépendances

6.5.1 Dépendances entre actes de langage

L'étude des dépendances entre actes de langage s'applique aux énoncés pour lesquels l'application des traitements des niveaux précédents fait apparaître plusieurs noyaux sémantiques correspondant à des actes de langage. Elle consiste à prendre en compte le contexte des actes précédents pour compléter la compréhension de l'acte concerné⁵. Deux traitements sont opérés :

- Le premier consiste à reconnaître les références, puis à les résoudre partiellement. Les références sont identifiables soit dans la représentation sémantique elle-même (dans l'exemple, les expressions « *y aller* » et « *d'ici* » ont produit dans la représentation sémantique des objets *objet_contexte* et *lieu_contexte*), soit enfin dans la catégorie syntaxique (dans une expression telle que « *cet hôtel* », c'est l'adjectif démonstratif « *cet* » qui indique qu'il y a référence à un objet du contexte). La résolution consiste à chercher dans le contexte quels objets peuvent sémantiquement correspondre à ces références. La connaissance syntaxico-sémantique est utilisée une fois de plus (en l'occurrence, seules les conditions sémantiques qu'elle contient sont utilisées). Ainsi, dans l'énoncé exemple, le lieu contextuel est la gare Saint Lazare⁶. Les conditions imposées à l'objet *Obj* dans la relation *est_dependant_de*(_, *Obj*, *aller*, *vers*) montrent que le seul objet du contexte qui peut sémantiquement correspondre au « *y* » de « *y aller* » est l'hôtel Caumartin (l'objet doit en effet posséder une propriété d'étiquette *identite*).

5. Cette mise en contexte utilise l'implication intuitionniste, l'un des connecteurs des formules manipulées par λ Prolog (formules de Harrop), cf. annexe B. Les objets des actes précédents sont dynamiquement mis en contexte pour la complétion de l'acte analysé.

6. Les énoncés du corpus PARISCORP (cf. note page 63) étaient censés correspondre à des requêtes d'un utilisateur situé dans la gare Saint-Lazare.

```

((info presence) (de (reservation [])
                    (de (chambre [])
                        (hotel [(nbetoiles (entier 2)),
                               (identite (nom "Caumartin"))])))

((interr. comment) (aller [(vers (hotel [(nbetoiles(entier2)),
                                         (identite (nom "Caumartin"))])),
                            (venant_de (gare [(identite (nom "StLazare"))]))]))

```

FIG. 6.13 – Représentation sémantique finale de l'énoncé exemple : formule logique.

Dans cette étude des dépendances entre actes de langage, les références ne posent pas de gros problèmes. Il est probable que les ambiguïtés seraient plus difficiles à lever dans le contexte d'un véritable dialogue. Pour valider l'approche de ces traitements, il faudrait pouvoir les évaluer sur de plus larges corpus.

- Le second traitement consiste à compléter les chaînes d'objets sur lesquelles portent les actes. Cette complétion utilise le concept de *segment droit* d'une chaîne d'objets défini précédemment (page 68).

Chaque proposition est considérée comme le contexte de la proposition suivante. Le principe fondamental de la complétion est que les objets d'une proposition peuvent être complétés par les segments droits des objets contextuels.

Par exemple, l'analyse de l'énoncé suivant distingue deux propositions :

« *je voudrais réserver une chambre à l'hôtel Caumartin combien ça coûte* » .

Proposition 1 :

```

((requete vouloir)
 (de (reservation [])
     (de (chambre []) (hotel [(identite (nom "Caumartin"))])))

```

Proposition 2 :

```

((interrogation quoi) (tarif []))

```

L'objet de la première proposition contient 3 segments droits. La connaissance sémantique permet les 3 complétions correspondantes (*tarif* est un sous-objet de *reservation*, *chambre* et *hotel*).

La proposition 2 peut donc être complétée en :

- 1 - ((interrogation quoi)
 (de (tarif []) (de (reservation [])
 (de (chambre []) (hotel [(identite (nom "Caumartin"))])))
)
)

 2 - ((interrogation quoi)
 (de (tarif []) (de (chambre [])
 (hotel [(identite (nom "Caumartin"))])))
)
)

- 3 - ((interrogation quoi)
 (de (tarif []) (hotel [(identite (nom "Caumartin"))]))
)

Si la deuxième proposition avait été « *c'est quoi l'adresse* », seule la troisième des 3 complétions eût été sémantiquement possible. A priori, l'heuristique consistant à prendre le plus grand segment droit possible semble raisonnable. Cependant actuellement, cette étude des dépendances entre actes reste théorique et n'a pas été soumise à l'épreuve de tests importants (l'évaluation principale à laquelle a été soumise LOGUS a été la campagne d'évaluation « par défi » (cf. page 115) et les énoncés comportant plus d'un acte y étaient rares.

6.5.2 Ébauche d'interprétation contextuelle

L'étude des dépendances entre actes de langage dans un même énoncé contient les éléments d'une ébauche d'interprétation des énoncés en contexte : résolution des références et complétion des chaînes d'objets. Il convient d'opérer également une complétion des actes de langage. Elle est évidente par exemple dans la succession de requêtes suivante :

- « *Quelle est la ligne de métro à prendre pour aller au Louvre* »
 « *Et la ligne de métro pour Notre-Dame* »

Quelques éléments de cette interprétation contextuelle ont été implémentés dans LOGUS suivant ces principes. Cependant, l'absence de corpus ne permet pas de développer l'approche envisagée ni d'évaluer précisément la pertinence de cette approche.

6.6 Conclusion

Dans la première version de LOGUS (LOGUS-I), l'analyse se déroule suivant deux grandes étapes, fondées sur des principes et des formalismes différents. Progressivement, la mise au point de LOGUS s'est accompagnée d'une augmentation du nombre de ces étapes, jusqu'au traitement très incrémental qui vient d'être présenté, avec un formalisme qui permet d'un bout à l'autre de l'analyse d'associer des arguments syntaxiques et sémantiques. Ces différents niveaux de composition généralisent les principes adoptés pour les analyses partielles classiques telles que la segmentation en *chunks* : ainsi, à chaque étape, seules les compositions « les plus évidentes » sont opérées, les problèmes plus complexes étant reportés à des stades ultérieurs de l'analyse.

Cette stratégie est censée permettre l'analyse d'énoncés qui correspondent à des niveaux de langue variés : les énoncés syntaxiquement et sémantiquement cohérents peuvent être traités entièrement à la fin du niveau 2, les énoncés disloqués ou mal reconnus nécessitant d'aller jusqu'à la détection des noyaux sémantiques.

À l'issue de la présentation qui vient d'être faite de l'approche choisie et de sa mise en œuvre, plusieurs questions méritent d'être discutées : la stratégie d'analyse peut-elle être considérée comme efficace par rapport aux objectifs initialement fixés ? Quels sont parmi les choix opérés ceux qui semblent les plus pertinents ? Ces deux points, ainsi que les développements et les applications possibles de LOGUS sont le sujet du dernier chapitre de ce mémoire.

Chapitre 7

Bilans et perspectives

Avant de se demander comment le système peut être amélioré ou quelles sont ses utilisations potentielles, on peut s'interroger sur ses performances et sur l'intérêt des solutions proposées par rapport aux objectifs initialement fixés. Les évaluations subies par LOGUS permettent de répondre partiellement à la première de ces deux interrogations.

7.1 Évaluations de LOGUS

Le tour d'horizon des paradigmes d'évaluation faits dans le chapitre 1 (cf. page 20) permet de juger des difficultés pour évaluer un système de CAP, surtout s'il s'agit d'un prototype non encore utilisé dans un système de DOHM. Les évaluations subjectives et celles de type « boîte noire » sont nécessairement exclues. Leur absence de généralité rend les évaluations de type ATIS impossibles et de toute façon leur faible pouvoir de diagnostic les rend peu intéressantes pour un système en cours de développement.

LOGUS n'est donc concerné que par les trois derniers paradigmes décrits dans le paragraphe qui leur a été consacré (§1.6.2) : DCR, « par défi » et PEACE. Actuellement, le système a subi deux évaluations qui correspondent aux deux premiers de ces trois paradigmes. Il doit participer prochainement à la campagne d'évaluation MEDIA fondée sur le troisième. Ces deux premières évaluations ont essentiellement permis de porter un diagnostic sur les points forts et les points faibles du système ; ce paragraphe tente d'en rendre compte.

La toute première version de LOGUS, prototype de LOGUS-I, a été soumise à une évaluation de type DCR (Antoine et al., 2000). L'objectif principal de cette évaluation était la validation de la méthodologie DCR et non l'évaluation du système ; celui-ci était en effet trop peu développé pour que le diagnostic porté puisse avoir un réel impact sur sa mise au point. Par exemple, l'évaluation a détecté un traitement insuffisant des réparations : le diagnostic était exact mais l'ensemble des traitements prévus de ces phénomènes n'était pas implémenté.

À l'inverse, la campagne d'évaluation « par défi » à laquelle a participé LOGUS-I est intervenue alors que le système commençait à être opérationnel. Elle a ainsi permis de porter un diagnostic sur les points faibles de LOGUS-I, dont les principaux résultats ont déjà été présentés (cf. page 81). Ce diagnostic a eu des conséquences importantes : il a conduit à une révision de l'ensemble du formalisme et à la construction de LOGUS-II.

Le but essentiel de ce paragraphe est d'essayer d'évaluer l'évolution du système en comparant les performances de LOGUS-I et LOGUS-II sur les tests de l'évaluation « par défi ». Mais, auparavant, il est normal de se demander ce que l'on peut attendre d'une telle comparaison.

7.1.1 Résultats comparés : précautions et intérêt

L'objectif de la campagne d'évaluation « par défi » a été de porter un diagnostic sur les points forts et les points faibles de chacun des systèmes testés et non d'en faire une comparaison directe ; leurs domaines et leurs tâches étaient de toute façon très différents.

Par ailleurs, les énoncés utilisés dans cette campagne sont des oraux simulés : ils ne peuvent prétendre être représentatifs de la langue utilisée par des utilisateurs éventuels du système. Par conséquent, l'image du système donnée par les séries de résultats n'est pas significative. La présentation des séries de chiffres qui correspondent aux résultats de LOGUS-I n'est donc que de peu d'intérêt.

Cependant, la grande diversité des énoncés tests fait que la plupart des phénomènes de la parole spontanée y sont présents, même si leur fréquence d'apparition est artificielle. Par ailleurs, les formes à peu près semblables des représentations sémantiques utilisées par LOGUS-I et LOGUS-II rendent possible une comparaison directe entre les résultats de l'analyse rendue par les deux systèmes pour les 1200 énoncés tests de l'évaluation.

S'il est donc risqué de tirer des conclusions des résultats chiffrés dans l'absolu, par contre les comparaisons entre les résultats des deux systèmes sur chaque type de phénomènes doivent pouvoir permettre de tirer quelques conclusions. On peut ainsi espérer avoir une estimation de l'importance des progrès réalisés et surtout, mieux cerner la nature des problèmes qui résistent à l'approche choisie.

Une critique majeure peut cependant être émise à l'encontre de cette étude comparative : en effet l'évaluation de LOGUS-II est effectuée à partir des énoncés-tests qui ont servi à évaluer LOGUS-I et à détecter les points faibles de ce premier système. On peut suspecter que les améliorations constatées sont d'autant plus importantes que les modifications apportées au système ont précisément cherché à corriger les erreurs constatées sur les tests utilisés et que, par conséquent, les écarts observés sont considérablement biaisés.

Malgré ces réserves, les comparaisons restent cependant intéressantes : en effet, si l'écart absolu entre les résultats obtenus par les deux systèmes sur chacune des séries des tests peut être contesté, les différences entre les écarts sont, quant à elles, significatives ; elles permettent de distinguer ce qui a pu être corrigé efficacement de ce qui pose réellement problème par rapport à l'approche choisie.

7.1.2 Résultats globaux

Les 1200 énoncés sont répartis en quatre séries dont chacune en comporte 300.

- Dans les séries 1 et 2, les énoncés tests sont obtenus en faisant subir à l'énoncé initial une ou deux modifications qui correspondent à des erreurs de reconnaissance ou à des phénomènes bien déterminés de la langue orale spontanée : incises, ellipses, mouvements des constituants, etc.

Par exemple, l'énoncé « *y a-t-il des restaurants ouverts le dimanche* » se voit dérivé en :
« *y a-t-il des hôtels euh non des restaurants voilà des restaurants ouverts le dimanche* »
(réparation)

ou en « *utile des restaurants ouverts le dimanche* » (erreur de reconnaissance)

Ainsi, les 600 énoncés que comportent ces deux séries permettent essentiellement de mesurer la robustesse du système face aux spécificités de l'expression orale et à un nombre réduit d'erreurs de la reconnaissance verbale.

- Les tests de la série 3 correspondent essentiellement à des variations sémantiques autour du thème de l'énoncé initial. Ainsi par exemple l'énoncé initial « *quels sont les horaires d'ouverture du Louvre* » se voit dérivé en « *entre le Louvre et le Grévin quel est celui qui ferme le plus tard* ». Cette série permet donc de mesurer les capacités du système à rendre compte de liens complexes entre objets et actes de langage : comparaisons, refus, confirmations, etc.
- Les tests de la série 4 respectent le sens général des énoncés initiaux mais les manifestations langagières de l'expression orale spontanée y sont poussées à l'extrême : très larges incises, changements de parcours syntaxiques et sémantiques, réparations, énoncés disloqués, etc. Par exemple, l'énoncé « *je voudrais réserver deux chambres pour ce soir à l'hôtel caumartin* » est dérivé en « *je voudrais réserver des chambres alors en fait j'appelle aussi pour un collègue et donc ben il en faudrait deux et donc c'est assez urgent ce serait pour ce soir à l'hôtel Caumartin* »

Les chiffres comparatifs des résultats obtenus par LOGUS-I et LOGUS-II sont indiqués dans le tableau de la figure 7.1 :

- Un « énoncé compris » suppose une représentation sémantique qui correspond à une modélisation correcte et complète des indices de l'acte de langage et de la chaîne d'objets sur laquelle il porte.
- Les énoncés sont classés dans la catégorie « compréhension incomplète » lorsque son sens général a été dégagé (les indices permettant d'inférer l'acte de langage et la chaîne d'objets à laquelle il s'applique) mais qu'il y a eu omission d'un élément non essentiel (l'une des propriétés d'un objet par exemple) ; autrement dit, la modélisation est correcte et incomplète et l'incomplétude correspond à un élément jugé non essentiel. Par exemple, l'énoncé « *et bien dans un rayon de cent mètres autour de la gare le moins cher toujours* » compris comme « *près de la gare et le moins cher* » a été classé dans cette catégorie.

Par rapport à LOGUS-I, LOGUS-II utilise des règles génériques, prétend distinguer les différentes propositions d'un énoncé et s'appuie sur des considérations syntaxiques après la segmentation en *chunks* pour affiner la compréhension. Par ailleurs, l'un des efforts essentiels lors de l'élaboration de LOGUS-II a été consacré à l'élargissement de la couverture sémantique : augmentation du nombre des objets et de leurs propriétés, élargissement de la reconnaissance et de la représentation dans le langage cible des actes de langage.

Les progrès enregistrés dans les résultats de la série 3 étaient les plus attendus ; ils traduisent à la fois un élargissement du domaine couvert par la compréhension et une amélioration de sa finesse.

La comparaison entre les deux systèmes sur les résultats des séries 1 et 2 indiquent seulement que cette extension du domaine n'a pas détérioré les performances obtenues initialement dans un domaine sensiblement plus restreint.

LOGUS-I	Séries 1 et 2	Série 3	Série 4	Total
Énoncés compris : en %	92.6	52.3	48	71.4
Compréhension incomplète : en %	3.5	16	16.7	9.9
% total	96.1	68.3	64.3	81.3
LOGUS-II	Série 1 et 2	Série 3	Série 4	Total
Énoncés compris : en %	93.7	73	61.7	80.5
Compréhension incomplète : en %	4.3	16.7	23.7	12.3
% total	98	89.7	85.4	92.8

FIG. 7.1 – Résultats comparés de LOGUS-I et de LOGUS-II

	Nombre		%	
	LOGUS-I	LOGUS-II	LOGUS-I	LOGUS-II
Substitution (objet, propriété, nature de l'acte)	32	6	12%	8%
Insertion (objet, propriété, nature de l'acte)	10	8	4%	11%
Omission (objet, propriété, nature de l'acte)	84	42	32%	59%
Liens absents	125	3	48%	4%
Liens erronés	11	12	4%	17%
Total	262	71	100%	100%

FIG. 7.2 – Nature des erreurs dans LOGUS-I et LOGUS-II

Les progrès observés dans les résultats de la série 4 sont selon nous très encourageants. En effet, dans des énoncés aussi disloqués, il était à craindre que la prise en compte d'éléments syntaxiques ne bloquât l'analyse ; les tests prouvent qu'il n'en est rien et il est probable que l'utilisation d'une analyse incrémentale en est la cause. Loin de contrarier l'analyse, les éléments syntaxiques aident à la reconnaissance des *noyaux sémantiques* (cf. 6.4.4) et permettent ainsi de dégager le sens général de la plupart de ces énoncés.

7.1.3 La nature des erreurs

Le tableau de la figure 7.2 précise la nature des erreurs observées suivant une terminologie adaptée à la forme des représentations sémantiques de LOGUS mais inspirée de celle adoptée pour les représentations sémantiques en schémas prédéfinis :

- Les substitutions, omissions et insertions peuvent concerner des propriétés, des objets ou même parfois des actes de langage. Les erreurs ne sont classées dans l'une de ces trois catégories que si l'énoncé comporte un ou plusieurs noyaux sémantiques principaux qui

correspondent à l'essentiel du message sémantique contenu dans l'énoncé (au sens d'un « expert humain »).

- Lorsque l'analyse laisse l'énoncé sous la forme de constituants indépendants (forme que l'on peut donc considérer comme incohérente), l'erreur correspondante est classée dans la catégorie des *liens absents*.
- Enfin, les *liens erronés* désignent des constructions sémantiques qui correspondent à des contre-sens.

Dans LOGUS-I, les erreurs correspondent le plus souvent à un échec global de l'analyse qui ne parvient pas à établir les liens sémantiques entre les différents *chunks* (ligne *Liens absents* : 48%). Dans LOGUS-II, ces échecs globaux de l'analyse sont considérablement moins fréquents ; les dernières étapes constituent un *passage en force* de la compréhension : les liens sémantiques sont établis avec des contraintes syntaxiques relâchées tant au niveau de l'ordre des constituants qu'au niveau de leur nature syntaxique. Le risque encouru avec une telle stratégie est évidemment de créer des erreurs dans les liens sémantiques ; celles-ci peuvent être classées comme des *liens erronés* mais elles peuvent aussi parfois correspondre à des *substitutions* ou à des *insertions*.

Les résultats du tableau font apparaître que la très grande majorité des erreurs de LOGUS-II correspondent à des omissions et que, si les substitutions, insertions et liens erronés représentent un pourcentage d'erreurs beaucoup plus important que dans LOGUS-I, leur nombre n'a cependant pas augmenté dans l'absolu. On peut donc en conclure que la compréhension forcée des dernières étapes n'a pas ou n'a que très peu créé de liens sémantiques erronés entre les constituants. La stratégie mise en œuvre semble donc efficace, même si elle conduit éventuellement à négliger une partie sémantiquement signifiante de l'énoncé.

7.1.4 Les causes des erreurs

Le tableau de la figure 7.3 correspond à l'analyse des causes d'erreurs sur 1000 énoncés de l'évaluation par défi¹.

- Les erreurs peuvent avoir pour cause les *erreurs de la reconnaissance* vocale simulées présentes dans les énoncés.
- Les *réparations locales* désignent les hésitations, reprises et répétitions limitées à un syntagme nominal, verbal ou adjectival.
- Le terme *organisation syntaxique et sémantique* recouvre à la fois les incises, les réparations dont la portée dépasse les syntagmes précédents (par exemple les faux-départs), les dislocations et d'une manière générale toutes les constructions induisant des discontinuités dans le parcours syntaxique ou sémantique.
- Le *pouvoir expressif de la langue cible insuffisant* regroupe les erreurs qui sont dues à un énoncé qui ne peut être traduit correctement dans la langue formelle définie dans le système.
- Le *lexique insuffisant* correspond aux erreurs dues à la présence dans l'énoncé de lexèmes qui sont absents du lexique de LOGUS ou qui y figurent avec une liste de définitions incomplète.

1. Deux cents énoncés de la série 4 ont été écartés. Leur complexité rend en effet difficile l'analyse des causes des erreurs les concernant.

	Nombre		Pourcentage des erreurs totales	
	LOGUS-I	LOGUS-II	LOGUS-I	LOGUS-II
Erreurs de reconnaissance	11	12	4%	17%
Réparations locales (à l'intérieur d'un syntagme)	19	9	7%	13%
Organisation syntaxique et sémantique	158	32	60%	44%
Pouvoir expressif de la langue cible insuffisant	41	7	16%	10.5%
Lexique insuffisant	33	13	13%	15.5%

FIG. 7.3 – Causes d'erreurs comparées dans LOGUS-I et LOGUS-II

Les résultats observés appellent deux remarques :

- L'influence des erreurs de reconnaissance n'a pas diminué entre les deux systèmes. La correction de certaines de ces erreurs n'est en effet envisageable que si les modules de compréhension et de reconnaissance interfèrent : LOGUS, dans sa version actuelle, ne peut remplacer un mot de la liste qu'il prend en entrée par un autre. Ainsi, si un mot qui représente un concept signifiant est substitué ou éliminé, l'analyse est presque nécessairement erronée ou incomplète. Dans l'état actuel de la technologie, les erreurs de reconnaissance constituent un obstacle majeur à la mise en œuvre d'une compréhension fine des énoncés.
- La cause d'erreurs majeure dans LOGUS-I comme dans LOGUS-II est la difficulté à établir les liens sémantiques entre les différents syntagmes. La variabilité de l'organisation syntaxique et sémantique des énoncés dans la langue naturelle en est la cause. L'hétérogénéité et l'importance de cette cause d'erreurs appellent une étude plus approfondie qui est présentée dans le paragraphe suivant.

Le tableau de la figure 7.4 propose un classement des erreurs désignées sous le terme générique de *organisation syntaxique et sémantique* dans le tableau précédent (figure 7.3).

- Les *incises* forment une classe particulière relativement facile à déterminer. Les erreurs qu'elles occasionnent sont nombreuses lorsqu'elles contiennent des éléments sémantiquement signifiants qui ne peuvent se composer avec les éléments situés à proximité. Les règles syntaxico-sémantiques les traitent difficilement. Seule l'étape des noyaux sémantiques permettent éventuellement d'en rendre compte.
- Les *auto-corrections* correspondent aux réparations de portée importante (il s'agit essentiellement des faux-départs). L'utilisation de critères syntaxiques dans la deuxième partie de l'analyse de LOGUS-II permet de bien mieux gérer ce type de phénomènes que dans LOGUS-I, même si le pourcentage des erreurs qu'elles occasionnent reste relativement élevé.
- Les *mouvements des constituants* désignent les dislocations diverses (par exemple : « *de la gare la plus proche comme pharmacie c'est laquelle* »). Le traitement de ces phénomènes

Organisation syntaxique et sémantique complexe	Incises	Auto-corrrections	Mouvements des constituants	Constructions syntaxiques complexes	Divers
LOGUS-I	22%	16%	32.5%	16.5%	13%
LOGUS-II	24%	12%	20%	44% (20% pour coordinations et négations)	0%

FIG. 7.4 – Les erreurs dues à la complexité de l'organisation syntaxique et sémantique de l'énoncé

est nettement plus efficace dans LOGUS-II.

- Les *constructions syntaxiques complexes* incluent en particulier certaines négations et coordinations de portée non évidente. Dans LOGUS-II, elles deviennent la cause d'erreurs la plus fréquente.
- La catégorie *divers* regroupe les erreurs considérées comme des *organisations syntaxiques ou sémantiques complexes* mais difficilement classées dans l'une des catégories précédentes.

Encore une fois, les pourcentages en eux-mêmes ne signifient pas grand chose puisque la fréquence de chaque phénomène dans le corpus est artificielle. Les comparaisons entre ces pourcentages permettent cependant de cerner la nature des problèmes les plus « durs », qui résistent aux progrès généralement enregistrés.

Si dans l'absolu, LOGUS-II obtient de meilleurs résultats que LOGUS-I sur chaque classe de phénomènes, ceux pour lesquels les progrès sont les moins nets sont les incises d'une part et la portée des coordinations et des négations d'autre part, deux types de phénomènes considérés généralement comme difficiles. Les stratégies d'analyse mises en œuvre lors de l'étape des règles syntaxico-sémantiques dans LOGUS-II semblent a priori contenir tous les éléments nécessaires au traitement du second d'entre eux : prise en compte de l'ordre des constituants, de leur contenu sémantique et des indices syntaxiques. Par contre, cette étape n'est pas particulièrement adaptée au traitement des incises : lorsqu'une incise contient des éléments épars du domaine de connaissance, elle crée des noyaux sémantiques dispersés, difficiles à relier sémantiquement au reste de l'énoncé et qui font courir un risque important de liens erronés. L'approche par les « noyaux sémantiques » demande sans doute à être raffinée pour parvenir à un meilleur traitement de ce type de phénomènes.

7.1.5 Erreurs de reconnaissance

Comme il a déjà été dit (cf. page 61), LOGUS a été conçu comme un module de compréhension indépendant du module de reconnaissance de la parole situé en amont dans le système. Les tests de l'évaluation « par défi » ne comportent qu'un nombre réduit d'erreurs de reconnaissance. Pour évaluer un peu plus sérieusement l'impact de ces erreurs sur la compréhension de LOGUS, nous avons fait un test avec un système grand public : le système ViaVoice de chez IBM.

Ce test ne peut avoir valeur de véritable évaluation pour plusieurs raisons :

- Si LOGUS était intégré dans un véritable système de DOHM, le lexique de la reconnaissance vocale serait lié au domaine concerné. Comme le système de reconnaissance choisi pour ce

test est destiné à des usages divers, son lexique est au contraire extrêmement vaste, ce qui n'est pas sans influence sur la nature de ses erreurs :

- « *c'est quelle heure* »/« *séquanais* »
- « *proche de la gare* »/« *proche de la guerre* »
- « *un carnet de tickets* »/« *incarner ticket* »
- Par ailleurs, lorsqu'on lit un énoncé, même en faisant des efforts, il est impossible de simuler parfaitement le rythme et la prononciation qui correspondent à une élocution spontanée.

Néanmoins, les résultats de cette expérience méritent d'être rapportés car ils ont permis quelques observations intéressantes sur le comportement de LOGUS face aux erreurs de la reconnaissance.

- Sur la centaine des énoncés choisis (pour leur diversité) dans les énoncés de l'évaluation « par défi », plus des $\frac{2}{3}$ ont fait l'objet d'au moins une erreur de reconnaissance. Beaucoup de ces erreurs concernent des mots courts et correspondent à des (quasi)-homonymies : « *et* »/« *est* », « *non* »/« *n'ont* », « *quel est* »/« *elle est* », « *s'est* »/« *c'est* », « *s'est* »/« *sait* »
- Environ la moitié des erreurs de reconnaissance (une quarantaine) portent sur des mots grammaticaux que l'on peut qualifier de « non-essentiels » : un expert humain peut les corriger aisément. À l'exception d'un « *et* » transformé en « *est* », ils n'occasionnent aucune erreur de compréhension de la part de LOGUS.
- En revanche, les quelques (6) erreurs observées sur des mots grammaticaux « essentiels » (« *non* » transformé en « *rond* » ou « *n'ont* », insertion d'un « *pas* », etc) provoquent (logiquement) une erreur de compréhension.
- Comme on pouvait l'espérer, la transformation d'un objet ou d'une propriété en un mot hors-domaine a pour conséquence l'omission du concept correspondant dans la chaîne d'objet rendue.
- La substitution d'un objet ou d'une propriété du domaine en un autre ne conduit pas nécessairement à une erreur :
 - « *à quelle heure ouvre le Louvre* » reconnu comme
« *à quelle heure Louvre le Louvre* »
donne en résultat
(*(interrogation quel) (de (horaire []) (musee [(identite (nom " Louvre"))]))*)),
 - « *les horaires d'ou d'ouvertures* » reconnu comme
« *les horaires doubles d'ouverture* » donne
(*de (horaire []) (ouverture [])*).
 - Par contre, « *des réductions pour le métro et le RER* » reconnu comme
« *des réductions pour le métros l'air horaires* » est compris comme
(*de (horaire []) (de (reduction []) (metro []))*)).
Seul, ce dernier exemple correspond à une véritable erreur de compréhension de LOGUS.

Apparemment, l'analyse faite dans LOGUS est suffisamment robuste pour faire face à la plupart des erreurs de reconnaissance rencontrées ; en particulier, LOGUS rend une représentation sémantique conforme à l'énoncé soumis même avec la plupart des mots grammaticaux oubliés ou substitués.

7.1.6 Conclusion

Si dans l'ensemble les résultats de LOGUS-I aux tests de l'évaluation « par défi » sont encourageants, les comparaisons entre les résultats de LOGUS-I et de LOGUS-II le sont également : elles semblent indiquer que le second système est sensiblement plus robuste que le premier et qu'il est capable de rendre compte d'énoncés sensiblement plus complexes.

Cependant, ces résultats ne permettent de juger ni des dernières étapes de l'analyse : détection d'actes de langage successifs et étude de leurs dépendances, interprétation contextuelle, ni de l'intérêt des informations sémantiques présentes dans la formule rendue et de la forme logique qui leur est donnée.

Par ailleurs, s'ils donnent une indication sur les points forts et faibles du système, ils ne peuvent suffire à comprendre ce qui, dans l'approche choisie, conduit à l'un ou l'autre de ces résultats.

7.2 Stratégies d'analyse : objectifs initiaux et perspectives

L'objectif initial qui a présidé à la conception de LOGUS était de faire des énoncés une analyse aussi fine et robuste que possible. Un autre objectif était que le système puisse être facilement réutilisé pour d'autres applications.

L'analyse mise en œuvre dans LOGUS se caractérise essentiellement par :

- la forme de la langue cible utilisée,
- le formalisme de représentation des constituants,
- une segmentation en *chunks* minimaliste,
- un système de règles par niveaux qui conduit à une analyse « en cascades ».

L'objet de ce paragraphe est d'essayer de cerner l'intérêt de ces propositions par rapport aux trois objectifs visés : robustesse, finesse et généricité. Il indique également quelques développements possibles du système.

7.2.1 Robustesse

Les principaux obstacles auxquels se heurte l'analyse des énoncés oraux spontanés sont la présence de réparations, une organisation syntaxique et sémantique souvent complexe et les erreurs de la reconnaissance vocale.

Stratégie d'analyse et réparations :

Le traitement des réparations dans LOGUS se fait à des niveaux différents, qui dépendent de leur nature et en particulier de leur portée :

- Lorsque le *reparandum* de la réparation est composé de mots grammaticaux, ceux-ci sont en général éliminés après la segmentation en *chunks* lors de l'opération de *filtrage* (cf. page 96). L'utilisation des *chunks minimaux* permet que ce filtrage soit non destructeur : comme aucun mot lexical ne peut être négligé, il ne fait pas courir le risque d'une perte d'informations sémantiques (cf. page 93).

- Lorsque les éléments réparés contiennent des mots considérés comme sémantiquement significatifs par le système, le traitement de la réparation est opéré par l'application de règles de composition analogues à celles utilisées pour les coordinations ; le formalisme en triplet de représentation des constituants permet une utilisation conjointe des indices syntaxiques et sémantiques. Le « niveau » auquel est traitée la réparation dépend de la convergence de ces indices. Par exemple, l'expression « *un ticket de métro non de bus* » peut être traitée par des règles de niveau 2 : les indices syntaxiques (répétition de la préposition « *de* ») et sémantiques (*metro* et *bus* appartiennent à une même famille sémantique) coïncident pour indiquer la portée de la réparation « *de métro* »/« *de bus* » et non « *un ticket de métro* »/« *de bus* ». Par contre, « *ticket de métro non un bus* » est traité au niveau 3 à cause de l'absence d'indice syntaxique.

La quasi-totalité des réparations dont la portée ne dépasse pas celle d'un groupe nominal font l'objet d'une analyse correcte. Cependant, les évaluations font nettement apparaître que plus la portée de la réparation est importante, plus le système peine à les traiter correctement : les faux-départs restent assez souvent une cause d'erreurs.

Si l'on ne peut pas prétendre que LOGUS traite correctement la totalité des réparations, il semble quand même que leur traitement soit l'un des points forts du système : l'utilisation de méthodes sélectives (cf. page 34) ne résoud pas ce problème et, par ailleurs, leur traitement par les règles syntaxico-sémantiques est une méthode beaucoup plus fine que la simple détection de motifs (cf. page 35). En effet, si les règles permettent de détecter aisément de tels motifs, elles permettent de s'appuyer également sur des critères syntaxiques et sémantiques. Les faux-départs se prêtent moins bien à ce traitement par règles car il n'est pas forcément simple de faire la distinction entre une succession « normale » de deux actes de langage et une correction du premier acte par le second.

Le traitement des réparations pourrait probablement être encore affiné en faisant intervenir dans la définition des règles un élément prosodique ; comme cela a déjà été dit au §6.1.2, il serait extrêmement facile d'ajouter un tel élément au formalisme de représentation des constituants (actuellement en triplet).

Stratégie d'analyse et organisation syntaxique et sémantique

L'oral spontané se caractérise par la souplesse des constructions syntaxiques et sémantiques : en particulier, les changements de parcours rendent difficile une analyse linéaire de l'énoncé.

Tous les systèmes basés sur une analyse par règles (comme par exemple TINA) sont confrontés à ce problème. En revanche, les analyses sélectives semblent mieux armées pour y faire face. Cependant, renoncer à dégager la structure générale de l'énoncé a un prix : les stratégies qu'elles utilisent ne peuvent permettre de rendre compte aisément d'une certaine complexité ; elles rencontrent des difficultés dès que l'énoncé comporte plus d'un acte de langage ou des objets coordonnés.

La stratégie d'analyse de LOGUS privilégie un développement linéaire de l'énoncé : la segmentation en *chunks* et les trois premiers niveaux de règles associent des éléments juxtaposés et dans un certain ordre. La dernière étape, centrée sur les noyaux sémantiques, tente de pallier les difficultés engendrées par les changements de parcours et les incises. Le caractère incrémental de l'analyse avec relâchement progressif des contraintes est un élément essentiel ; il permet une stratégie mixte : il favorise l'analyse de l'énoncé lorsque celui-ci suit un développement « normal » et linéaire sans

pour autant le rejeter ou le soumettre à un traitement totalement différent en cas de problème. Ainsi, la stratégie d'analyse « en cascades » avec un relâchement progressif des contraintes présente l'avantage d'être adapté aussi bien à des énoncés syntaxiquement corrects qu'à une langue proche d'un *pidgin* (cf. page 32).

Stratégie d'analyse et erreurs de reconnaissance :

Comme dans la plupart des systèmes (cf. page 29), les erreurs de reconnaissance sont traitées « en creux » par LOGUS. Seule la robustesse du système permet réellement d'y faire face. Le principe général adopté étant en effet de « composer ce qui fait sens », on peut espérer que les remplacements de mots dus à des erreurs de reconnaissance créent des îlots sémantiquement incompatibles avec le reste de l'énoncé.

Il va de soi que l'application de tels principes doit impliquer dans la stratégie de dialogue un retour systématique à l'utilisateur de ce qui a été compris de son message. De tels retours sont d'ailleurs permanents dans un dialogue Homme-Homme où les interlocuteurs se manifestent en permanence qu'ils s'écoutent et se comprennent. Il convient en particulier de signaler autant que possible à l'utilisateur que son énoncé a été mal reconnu ou mal compris. À cet effet, LOGUS garde une trace des mots sautés ou négligés.

Par ailleurs, comme LOGUS n'est pas lié à un module de reconnaissance particulier, son lexique et celui de la reconnaissance ne coïncident pas. Le système est donc normalement confronté à des mots absents de son lexique, en particulier à cause des erreurs du module de reconnaissance.

Dans LOGUS, un tel *mot inconnu* donne lieu à une définition particulière, du type $\langle c_inconnu, r_inconnu, neutre \rangle$, avec une catégorie syntaxique et un rôle sémantique spécifiques. Au niveau de la segmentation en *chunks*, il va essentiellement fonctionner comme une barrière : deux mots situés de part et d'autre ne pourront pas appartenir à un même *chunk* minimal. Par ailleurs, cette segmentation permet de présumer de certaines catégories syntaxiques de mots inconnus. Par exemple, un pronom personnel situé devant un mot inconnu et non précédé d'un verbe permet d'induire à peu près certainement que le mot inconnu est un verbe. Cependant, il convient de ne pas oublier que la présence des réparations dans la langue orale ne permet aucune certitude (il est vrai que les erreurs possibles de la reconnaissance font que tous les mots de l'énoncé restent plus des hypothèses que des certitudes).

La barrière des mots inconnus est maintenue aux deux premiers niveaux de l'utilisation des règles syntaxico-sémantiques de la deuxième étape. La suppression des mots inconnus participe par contre de l'assouplissement des contraintes lors de troisième étape. Par ailleurs, un petit nombre de règles syntaxico-sémantiques font directement intervenir les mots inconnus (lorsqu'ils précèdent des marques d'hésitation ou de reprise).

Le traitement des mots inconnus et des erreurs de reconnaissance reste très rudimentaire. Compte tenu de l'importance du problème, la nécessité de traitements plus sophistiqués risque de se faire sentir pour une utilisation de LOGUS dans un système de dialogue réellement opérationnel. Le petit bout d'essai fait avec la reconnaissance vocale grand public suggère quelques pistes possibles ; par exemple, il serait simple de mettre sous la même entrée du lexique un certain nombre d'homonymes : « *et* »/« *est* », « *non* »/« *n'ont* », etc. « *s'est* »/« *c'est* ». On peut présager en effet que les règles syntaxico-sémantiques sont plus efficaces pour choisir entre les différentes définitions de ces entrées que les modèles de langage bigrammes ou même trigrammes utilisés par

la reconnaissance vocale (cf. page 8).

Conclusion

Concernant la robustesse de LOGUS, les évaluations donnent des indications encourageantes. Selon nous, son efficacité relative est due à la fois au caractère incrémental de l'analyse et à l'interpénétration des indices syntaxiques et sémantiques.

Des progrès sont bien sûr nécessaires mais le formalisme et la stratégie d'analyse devraient le permettre. L'introduction d'indices prosodiques semble simple (mais la modification des règles afin de les exploiter l'est certainement beaucoup moins). Par contre, la question de savoir comment exploiter des indices statistiques est totalement ouverte : introduction de probabilités dans les règles ? utilisation de critères statistiques pour l'écriture même de ces règles ? etc ? Comme il a déjà été dit, il s'agit certainement d'un axe de recherche intéressant.

7.2.2 Finesse

La finesse de la compréhension a été définie (page 61) comme la capacité du système à rendre compte le plus précisément possible du sens de cet énoncé. Le choix fait dans LOGUS a été celui d'une représentation sémantique composée de deux éléments : une chaîne d'objets qui représente les objets du domaine concerné (le *contenu propositionnel* de l'énoncé), un prédicat qui contient les intentions du locuteur sur les objets concernés (la *force illocutoire* de l'énoncé).

À côté du choix de la langue cible, la stratégie d'analyse mise en place est également un composant de la finesse : elle essaie de prendre en considération tous les éléments de l'énoncé, en abandonnant seulement ceux qui semblent incompatibles avec l'ensemble des autres (certains mots grammaticaux à l'issue de la segmentation, ensuite et progressivement d'autres éléments).

Pour au moins deux raisons, les évaluations ne permettent pas de savoir si l'objectif de finesse est réellement atteint :

1. Quelques éléments pour la compréhension en contexte de dialogue ont déjà été présentés (page 113). La résolution des références et la complétion des énoncés elliptiques reposent sur la donnée des objets et des actes de langage des échanges verbaux précédents et elles utilisent le formalisme des chaînes d'objets.

Cependant, l'interprétation de l'énoncé est encore actuellement incomplète, en particulier pour ce qui concerne les intentions du locuteur (la *force illocutoire* de l'énoncé). Actuellement en effet, LOGUS ne rend qu'une représentation des indices linguistiques de la nature des actes de langage. Les intentions réelles de l'utilisateur, souvent implicites ou exprimées sous forme indirecte, doivent se déduire de ces indices, de la nature des objets sur lesquels porte la requête et, bien sûr, du contexte du dialogue.

Les progrès sont freinés par le manque de corpus disponibles sur lesquels travailler. LOGUS doit participer à la campagne d'évaluation MEDIA (cf. page 24) ; une évaluation de la compréhension en contexte de dialogue y est prévue et des corpus vont être recueillis par la méthode du magicien d'Oz (la tâche doit porter sur la réservation hôtelière). Cette campagne devrait donc donner l'occasion de compléter le développement du système.

2. Les évaluations mesurent la distance entre la représentation obtenue et la représentation attendue. La définition donnée (page 63) de ce dont doit rendre compte la modélisation du

sens (donc de ce que l'on peut « attendre » de la compréhension) a été reliée à la notion de « *sens utile* ». Si cette définition a le mérite d'être simple, on peut lui reprocher d'être quelque peu autoréférentielle : l'« *utile* », c'est ce qui peut servir. Par conséquent, pour juger de la pertinence du choix de ce qui a été retenu comme utile, il est nécessaire d'intégrer le système de compréhension à la gestion du dialogue et d'évaluer la pertinence de ce dialogue. Pour pouvoir juger de la qualité de la langue cible, en particulier de la forme et du contenu des informations qu'elle véhicule, il conviendrait donc de construire une logique et une stratégie de dialogue qui puissent en faire usage, ce qui suppose une manipulation des formules (ou des graphes conceptuels) rendues par LOGUS. Le développement du système nécessite donc également des travaux (ambitieux) de ce type.

7.2.3 Généricité

Il est probable que l'objectif d'un système de compréhension de la parole hors d'un domaine spécifique est inaccessible à court ou moyen terme. Il est probable également que les systèmes de compréhension à but pragmatique sont d'autant plus efficaces qu'ils sont conçus pour une application spécifique dans un domaine particulier. L'élaboration de LOGUS a pour but de mettre en œuvre et d'expérimenter un formalisme et quelques principes généraux. Pour prouver l'intérêt de l'approche, le système se doit d'être efficace ; si la qualité de l'analyse est la composante essentielle de cette efficacité, une autre composante importante est la facilité à pouvoir adapter le système à un autre domaine d'application. Celle-ci est actuellement considérée comme un enjeu majeur (Minker, 1999). La généricité et la réutilisabilité des formalismes et des traitements ont donc constamment orienté les différents choix.

Ce paragraphe étudie le résultat de ces efforts et les transformations qu'il faudrait apporter au système pour pouvoir l'utiliser dans une autre application.

Lexique

Le lexique n'est évidemment pas générique. Les lexèmes qui y figurent proviennent de corpus relatifs à l'application : la couverture lexicale serait donc très incomplète pour tout autre domaine. Par ailleurs, les définitions données aux lexèmes comportent des éléments du langage cible, lui-même conçu pour traiter des objets du domaine : par exemple, le mot « *étoile* » figure dans le lexique avec des définitions qui ne sont relatives qu'aux catégories des hôtels ou des restaurants.

Or, la fabrication d'un lexique est une tâche très lourde, d'autant que la nature des définitions choisies fait que celles-ci peuvent difficilement être générées automatiquement ou être directement dérivées de lexiques existants. On peut donc penser qu'il s'agit là d'un premier obstacle à l'utilisation de LOGUS.

Cependant, l'ampleur de la tâche demandée pour transformer le lexique actuel pour une autre application demande à être relativisée ; en effet, la complexité des définitions concerne essentiellement les mots grammaticaux. Or, ces mots ne diffèrent que très peu d'une application à une autre. La plupart des définitions qui s'appliqueraient à un autre domaine sont celles de mots lexicaux, essentiellement verbes, adjectifs, noms communs et noms propres. La définition de ces mots est généralement très simple ; par exemple, le lexique de LOGUS-II donne une seule définition du mot « *hôtel* » qui correspond à $\langle \text{nomc, obj_acte, (hotel [])} \rangle$. Ainsi, si une modification du domaine de la tâche implique obligatoirement une révision

du lexique, celle-ci porte essentiellement sur ce type de mots ; leurs définitions dépendent directement du modèle des objets et propriétés du domaine.

Règles

Les règles utilisées lors de l'étape de la segmentation en *chunks* sont complètement génériques. Les compositions sont contenues dans les définitions du lexique.

Les règles syntaxico-sémantiques utilisées lors de l'étude des dépendances entre *chunks* s'appuient sur des connaissances sémantiques spécifiques au domaine. Malgré tout, le formalisme adopté dans LOGUS-II permet de concilier une certaine généralité des règles et l'utilisation de ces connaissances spécifiques. On a vu en effet (cf. page 101) que la définition de ces règles faisaient appel à la catégorie syntaxique et au rôle sémantique des constituants d'une part et aux prédicats de la connaissance sémantique du domaine d'autre part. Comme les étiquettes des propriétés interviennent dans la définition des rôles sémantiques, un changement de domaine peut impliquer la révision de quelques règles. Cependant, à condition que la connaissance syntaxico-sémantique du domaine soit construite à partir des relations imposées (*est_sous_objet_de*, *est_propriete_de*, etc, cf. page 97), la très grande majorité d'entre elles n'a pas à être modifiée.

Langage cible et connaissance sémantique

L'étude des dépendances syntaxico-sémantiques entre les objets et leurs propriétés est évidemment entièrement dépendante du domaine et un changement exige une reconstruction complète.

Il est probable d'ailleurs que si la tâche visée est l'interrogation d'une base de données, cette reconstruction devrait pouvoir s'appuyer fortement sur cette base de données, voire en être dérivée presque automatiquement, selon des méthodes qui restent à définir.

Pour adapter LOGUS à une autre application, la construction des dépendances syntaxico-sémantiques devrait probablement être le premier des travaux à effectuer : elle conditionne les modifications à apporter au langage cible et la révision du lexique (les traductions sémantiques sont des éléments du langage cible). La façon d'adapter LOGUS à un autre domaine d'application pourrait faire l'objet d'études plus approfondies.

7.3 Interrogations existentielles

Le choix de terminer ce mémoire par un tel paragraphe peut sembler paradoxal puisque la question posée remet en cause l'intérêt même de ce travail.

Le système proposé n'est qu'un élément d'un système de dialogue Homme-Machine et sa conception repose sur des hypothèses concernant particulièrement les modules amont et aval qu'il convient d'essayer d'explicitier. Dans quelle mesure et dans quelles conditions sont-elles vraisemblables ? À court terme, quelles sont les applications réelles possibles d'un système de compréhension de ce type ?

À la conférence annuelle de TALN 2003, au cours d'une table ronde, Jean Véronis a dénoncé l'*ubris*² dont a fait preuve la communauté TALN avec ses grandes ambitions déçues telles que la description complète des règles du langage naturel ou la traduction automatique, comparables à la

2. Chez les anciens Grecs, l'*υβρις* désigne la démesure, l'ambition démesurée, une transgression contre les normes imposées par la condition humaine. C'est un crime puni de mort par les dieux.

recherche de la pierre philosophale chez les alchimistes. La conception de LOGUS ne participe-t-elle pas, elle aussi, de cette ambition démesurée ?

L'un des postulats sur lesquels repose LOGUS est qu'il est possible et utile de faire une compréhension relativement détaillée des énoncés, après que le signal vocal a été traité par un module de reconnaissance de la parole. Un tel postulat sous-entend que les modules de reconnaissance délivrent un message à peu près intelligible. Or, actuellement, la reconnaissance de la parole est le maillon faible des systèmes vocaux ; si en dictée vocale, ses performances sont satisfaisantes, il n'en est pas de même dans une ambiance bruitée, en situation « naturelle » de dialogue. Les différences de ton, les bruits de fond et les efforts des locuteurs pour se faire entendre dans un fond sonore se conjuguent pour perturber le signal vocal et par conséquent, pour diminuer la qualité de la reconnaissance vocale ; d'après S. Oviatt, ces détériorations des performances de la reconnaissance dans ce type de situations sont souvent sous-estimées (Oviatt, 2000).

Par ailleurs, après avoir connu une ascension impressionnante, les progrès de la reconnaissance de la parole se sont sensiblement ralentis et tendent à se rapprocher d'un optimum relatif aux conditions technologiques actuelles.

Dans l'application actuellement envisagée, un dialogue interactif pour l'interrogation téléphonique d'une base de données, la reconnaissance de la parole se trouve confrontée à des difficultés cumulées : multiplicité des locuteurs, ambiance bruitée, transmission téléphonique. Dans ces conditions, avec un taux d'erreurs sur les mots qui risque d'approcher les 30%, on peut se demander si une compréhension fine des intentions des locuteurs et la gestion d'un dialogue « naturel » ne sont pas du domaine de l'utopie.

Selon nous, ces interrogations ne diminuent pas l'intérêt des recherches sur la compréhension de la langue orale spontanée.

- Tout d'abord, en dépit des difficultés liées à la reconnaissance vocale, des systèmes de DOHM sont d'ores et déjà considérés comme opérationnels. Pour connaître les limites de leur usage et leurs capacités tant du point de vue de la tâche que de celui de l'interactivité, il semble indispensable de tester jusqu'où leur développement est possible, en améliorant, autant que faire ce peut, chacun de leurs composants.

Dans le cas particulier de LOGUS, l'analyse incrémentale fait que le système peut s'adapter à différents niveaux de normalité du message qu'il a à analyser. Cette faculté permet de faire face à l'inconnue de l'importance des erreurs de reconnaissance ; seul l'usage permettrait de savoir quelle est la granularité optimale de la compréhension, dans des conditions réelles d'utilisation et de déterminer si la finesse de certaines règles est excessive.

- Par ailleurs, le dialogue oral interactif Homme-Machine n'a pas pour seule application l'interrogation téléphonique d'une base de données. L'on peut songer par exemple à la commande interactive d'un robot domestique destiné à aider une personne âgée ou handicapée. Dans un tel contexte, la tâche de la reconnaissance vocale est grandement facilitée : absence de media tel que le téléphone, reconnaissance mono-locuteur et ambiance très probablement moins bruitée. De plus, le caractère fermé mais cependant relativement vaste du domaine concerné semble bien adapté aux capacités de LOGUS.
- Enfin, on peut avancer que certains principes généraux d'un système tel que LOGUS pourraient être utilisés pour la compréhension de textes écrits dans un domaine particulier. Comme la couverture syntaxique du TAL n'est jamais totale, la pragmatique peut venir au secours de l'analyse syntaxique comme nous l'avons proposé pour LOGUS. Il va de soi ce-

pendant qu'une utilisation des principes de LOGUS dans un tel contexte pourrait utiliser un modèle syntaxique plus riche et plus précis.

L'importance que prendra le DOHM dans la vie courante à court ou moyen terme est actuellement inconnue. Son développement nécessite des efforts importants, dans différents domaines : l'interprétation des messages envoyés par les utilisateurs est l'un d'entre eux. Le travail qui vient d'être présenté se veut une participation à ces efforts.

Troisième partie

Annexes

Annexe A

Grammaires de Montague et λ -calcul

L'application des grammaires au langage naturel débute au début des années 50 avec les travaux de Chomsky qui y affirme l'indépendance de la syntaxe par rapport à la sémantique et par conséquent, l'autonomie de la syntaxe. Son but est de proposer des règles qui permettent de décrire les phrases grammaticales de la langue ; ces règles doivent permettre de connaître la « structure profonde » des énoncés grammaticaux. Actuellement, si les formalismes existants ne permettent jamais de couvrir la totalité des constructions syntaxiques rencontrées, ils savent rendre compte des constructions classiques. Cependant, le problème demeure de la construction de la représentation sémantique correspondante car en général, dans la plupart de ces formalismes, les liens entre la structure syntaxique et la sémantique ne sont pas clairement explicités. La recherche de la construction d'une représentation sémantique des énoncés liée à leur analyse syntaxique a réellement débuté avec les grammaires de Montague, liées aux grammaires catégorielles et au calcul de Lambek.

Dans la sémantique de Montague, la représentation sémantique des énoncés est une formule de la logique du premier ordre où les déterminants correspondent à des quantificateurs. Ainsi par exemple la représentation sémantique de l'énoncé « *Médor poursuit un chat* » est : $(\exists x ((chat\ x) \wedge (poursuivre\ Medor\ x)))$. Le calcul de Lambek permet un lien direct avec la sémantique de Montague : le calcul des séquents qui permet de démontrer la grammaticalité d'un énoncé correspond à un λ -terme simplement typé qui représente sémantiquement cet énoncé.

A.1 Grammaires catégorielles

Les premières grammaires catégorielles apparaissent dans les années 50-60 avec les grammaires AB (Bar-Hillel, 1964; Retoré, 2000). Dans ces grammaires, on se donne un ensemble fini P de catégories de base, par exemple $\{sn, nom_c, S\}$ (sn pour syntagme nominal et nom_c pour nom commun) où S joue un rôle particulier, analogue à celui joué par le symbole non-terminal initial des grammaires génératives. On définit inductivement l'ensemble des catégories par :

1. Les catégories de base sont des catégories.
2. Si C_1 et C_2 sont des catégories, alors $C_1 \setminus C_2$ et C_2 / C_1 sont des catégories.

Un lexique associe à chaque mot de la langue une ou plusieurs catégories. Les deux règles d'association entre les catégories correspondent à des règles de calcul sur les fractions (produit non

commutatif) :

1.

$$x (x \backslash y) \rightarrow y$$

2.

$$(y/x) x \rightarrow y$$

La règle 1 signifie qu'un mot de catégorie $(x \backslash y)$ peut prendre à sa gauche un mot de catégorie x pour former un composant de catégorie y et la règle 2 qu'un mot de catégorie (y/x) peut prendre à sa droite un mot de catégorie x pour former un composant de catégorie y .

Le langage engendré par cette grammaire est l'ensemble des suites de mots $m_1 m_2 \dots m_n$ tel que, pour chaque mot m_i , il existe une catégorie c_i telle que $c_1 c_2 \dots c_n \rightarrow^* S$. Ainsi par exemple, l'énoncé « *Pierre mange une pomme* » est reconnu grammatical par :

« Pierre »	« mange »	« une »	« pomme »	
sn	$sn \backslash (S/sn)$	sn/nom_c	nom_c	\rightarrow
sn	$sn \backslash (S/sn)$		sn	\rightarrow
	S/sn		sn	\rightarrow
		S		

A.2 Grammaires de Montague et λ -termes

La sémantique de Montague est une logique extensionnelle utilisée par Montague pour représenter la partie « référentielle » d'un fragment de la langue anglaise. La représentation sémantique des énoncés est une formule de la logique des prédicats du premier ordre. Par exemple, la formule logique qui correspond à l'énoncé « *Pierre rêve* » est $(reve\ Pierre)$ ou $reve$ est un prédicat d'arité 1 et $Pierre$ une constante. Les déterminants correspondent à des quantifications comme dans l'énoncé « *Pierre mange une pomme* » traduit par $\exists x ((pomme\ x) \wedge (mange\ Jean\ x))$ où $pomme$ est un prédicat d'arité 1 dont l'interprétation est $(pomme\ x)$ si et seulement si x est une pomme et où $mange$ un prédicat d'arité 2 d'interprétation $(mange\ x\ y)$ si et seulement si x mange y).

A.2.1 Curry-Howard

La correspondance de Curry-Howard établit un parallèle entre les types et les formules d'une part et entre les termes et les preuves d'autre part. Entre les λ -termes simplement typés et les preuves de la logique intuitionniste minimale (ayant le connecteur \rightarrow , l'implication intuitionniste, comme seul connecteur), la correspondance est même un isomorphisme.

A.2.2 Calcul de Lambek

À la fin des années cinquante, Lambek (Lambek, 1958) montre que l'on peut formaliser les grammaires catégorielles sous forme logique. La dérivation d'un énoncé correspond à une preuve dans un calcul séquentiel. Ce calcul répond à un certain nombre de propriétés.

Il est non commutatif : les hypothèses ne permutent pas entre elles, ce qui implique une prise en compte de l'ordre des ressources (donc de l'ordre des mots).

Il est linéaire (Retoré, 1996), au sens de la logique du même nom (Girard, 1987) : il n'y a pas de règles permettant d'ajouter ou de supprimer des hypothèses (règles d'affaiblissements ou de contractions). On a donc un calcul sensible « aux ressources » : chaque hypothèse doit être consommée une et une seule fois.

Il est intuitionniste : on n'autorise qu'une seule formule dans la partie droite des séquents. Un séquent peut être considéré comme une fonction qui consomme des ressources pour en produire une autre.

C'est ce caractère intuitionniste qui permet un lien direct entre ce calcul et la sémantique de Montague.

A.2.3 Les λ -termes simplement typés

Les types sont définis inductivement à partir d'un ensemble \mathcal{T} de types élémentaires :

1. Les types élémentaires sont des types.
2. Si a et b sont des types, alors $(a \rightarrow b)$ est un type.

Les termes du λ -calcul permettent la représentation des fonctions par les règles qu'ils représentent (Lallement, 1990). Si \mathcal{C}_t et \mathcal{V}_t désignent respectivement les identificateurs de constantes et de variables de type t , les règles de formation des λ -termes simplement typés correspondent aux trois règles de grammaire suivantes :

1. $\Lambda_t ::= \mathcal{C}_t | \mathcal{V}_t$: les constantes et les variables sont des λ -termes de type t .
2. $\Lambda_{t_1 \rightarrow t_2} ::= \lambda \mathcal{V}_{t_1}. \Lambda_{t_2}$: si x est une variable de type t_1 et L un λ -terme de type t_2 , alors $\lambda x.L$ est un λ -terme de type $(t_1 \rightarrow t_2)$ (appelé *abstraction*).
3. $\Lambda_{t_2} ::= (\Lambda_{t_1 \rightarrow t_2} \Lambda_{t_2})$: si L_1 et L_2 sont des λ -termes de types respectifs $t_1 \rightarrow t_2$ et t_2 , alors $(L_1 L_2)$ est un λ -terme de type t_2 . Il est appelé *application* de L_1 à L_2 .

Les λ -termes sont classiquement munis des trois équivalences suivantes :

- α -équivalence : elle correspond au renommage des variables.

$$\lambda x.E \equiv_{\alpha} \lambda y.E[x \leftarrow y]$$

La condition pour que ce renommage soit licite est que y n'appartienne ni aux variables libres ni aux variables liées de E .

- β -équivalence : elle correspond à l'application d'une fonction à l'un de ses arguments :

$$(\lambda x.(E) F) \equiv_{\beta} E[x \leftarrow F]$$

avec les variables libres de F non-liées dans E .

- η -équivalence : une fonction est déterminée par l'image qu'elle donne d'un argument quelconque (extensionnalité fonctionnelle)

$$\lambda x.(E x) \equiv_{\eta} E$$

à condition que x ne soit pas une variable libre de E .

La β -convertibilité de deux termes simplement typés est décidable. Si les λ -termes simplement typés n'ont pas la propriété de la complétude calculatoire, ils sont normalisables et même fortement normalisables (on peut atteindre leur forme normale par n'importe quelle suite de réductions).

mot	cat. synt.	type	λ -terme	Id.
« <i>Pierre</i> »	sn	$(e \rightarrow t) \rightarrow t$	$\lambda P.(P \text{ Pierre})$	Λ_P
« <i>dort</i> »	sn\s	$e \rightarrow t$	$\lambda x. (\text{dort } x)$	Λ_D
« <i>mange</i> »	(sn\s)/sn	$((e \rightarrow t) \rightarrow t) \rightarrow e \rightarrow t$	$\lambda C \lambda x.(C (\text{mange } x))$	Λ_M
« <i>une</i> »	sn/nc	$(e \rightarrow t) \rightarrow (e \rightarrow t) \rightarrow t$	$\lambda P. \lambda Q. (\exists x (P x) \wedge (Q x))$	Λ_U
« <i>pomme</i> »	nc	$(e \rightarrow t)$	$\lambda x. (\text{pomme } x)$	Λ_{Po}

A.2.4 Calcul de Lambek et sémantique de Montague

On définit les deux types élémentaires e , le type des entités et t , le type des énoncés (valeur de vérité). Ainsi par exemple, $((e \rightarrow t) \rightarrow (e \rightarrow (e \rightarrow t)))$ est un type. Pour simplifier l'écriture, le parenthésage est sous-entendu à droite, et le type précédent peut s'écrire $((e \rightarrow t) \rightarrow e \rightarrow e \rightarrow t)$. Un terme prédicatif d'arité 4 a donc par exemple pour type $(e \rightarrow e \rightarrow e \rightarrow e \rightarrow t)$ et un terme fonctionnel d'arité 3 pour type $(e \rightarrow e \rightarrow e \rightarrow e)$.

À chaque définition d'un mot m du langage est associé un type et un λ -terme. Par exemple, le nom propre « *Pierre* » est un syntagme nominal. Le λ -terme qui lui est associé correspond à l'ensemble des prédicats d'arité 1 qui peuvent être appliqués à l'entité (de type e) *Pierre* pour former la représentation sémantique de l'énoncé : $\lambda P.(P \text{ Pierre})$. Comme P est un prédicat de type $(e \rightarrow t)$, il s'agit d'une abstraction de type $((e \rightarrow t) \rightarrow t)$.

La grammaticalité d'un énoncé est démontrée par un calcul de la logique intuitionniste, grâce au calcul de Lambek. Grâce à l'isomorphisme de Curry-Howard, ce calcul des séquents correspond à un λ -terme simplement typé. La β -réduction de ce λ -terme permet d'obtenir la représentation sémantique de l'énoncé.

A.2.5 Exemples

La figure A.2.5 donne les catégories syntaxiques, les types et les λ -termes associés aux cinq mots des deux énoncés « *Pierre dort* » et « *Pierre mange une pomme* ».

« *Pierre dort* »

La grammaticalité de cet énoncé est assurée directement par le séquent :

$$\frac{\text{Pierre} \vdash \text{sn} \quad \text{dort} \vdash \text{s/sn}}{\text{Pierre, dort} \vdash \text{s}}$$

Appliquée à un calcul sur les types, cette démonstration devient :

$$\frac{\text{Pierre} \vdash (e \rightarrow t) \rightarrow t \quad \text{dort} \vdash e \rightarrow t}{\text{Pierre, dort} \vdash t}$$

Le λ -terme qui correspond à la représentation sémantique de l'énoncé s'obtient par β -réduction de la composition des λ -termes qui correspondent à cette démonstration :

$$\begin{aligned} (\Lambda_P \Lambda_D) &\equiv_{\beta} (\lambda P.(P \text{ Pierre}) \lambda x. (\text{dort } x)) \\ &\equiv_{\beta} (\lambda P.(P \text{ Pierre}) \text{ dort}) \\ &\equiv_{\beta} (\text{dort Pierre}) \end{aligned}$$

« Pierre mange une pomme »

La preuve suivante démontre la grammaticalité de l'énoncé.

$$\begin{array}{c}
 \frac{\text{Pierre} \vdash sn \quad \frac{\text{mange} \vdash (sn \setminus s) / sn \quad \frac{\text{une} \vdash sn / nomc \quad \text{pomme} \vdash nomc}{\text{une, pomme} \vdash sn}}{\text{mange, une, pomme} \vdash sn \setminus s}}{\text{Pierre, mange, une, pomme} \vdash s} \\
 \\
 \frac{\text{Pierre} \vdash (e \rightarrow t) \rightarrow t \quad \frac{\text{mange} \vdash \frac{((e \rightarrow t) \rightarrow t)}{\rightarrow e \rightarrow t} \quad \frac{\text{une} \vdash \frac{(e \rightarrow t) \rightarrow \quad \text{pomme} \vdash (e \rightarrow t)}{(e \rightarrow t) \rightarrow t}}{\text{une, pomme} \vdash (e \rightarrow t) \rightarrow t}}{\text{mange, une, pomme} \vdash e \rightarrow t}}{\text{Pierre, mange, une, pomme} \vdash t}
 \end{array}$$

Le λ -terme associé à cette démonstration s'obtient par la composition qui correspond à la preuve : $(\Lambda_P (\Lambda_M (\Lambda_U \Lambda_{Po})))$ par β -réduction :

$$\begin{aligned}
 & (\Lambda_P (\Lambda_M (\lambda P. \lambda Q. (\exists p (P p) \wedge (Q p)) \lambda x. (pomme x)))) \\
 \equiv_{\beta} & (\Lambda_P (\Lambda_M \lambda Q. (\exists p (pomme p) \wedge (Q p)))) \\
 \equiv_{\beta} & (\Lambda_P (\lambda C \lambda x. (C (mange x)) \lambda Q. (\exists p (pomme p) \wedge (Q p)))) \\
 \equiv_{\beta} & (\Lambda_P (\lambda x. (\lambda Q. (\exists p (pomme p) \wedge (Q p)) (mange x))) \\
 \equiv_{\beta} & (\Lambda_P \lambda x. (\exists p (pomme p) \wedge (mange x p))) \\
 \equiv_{\beta} & (\lambda x. (x Pierre) \lambda x. (\exists p (pomme p) \wedge (mange x p))) \\
 \equiv_{\beta} & \exists p (pomme p) \wedge (mange Pierre p)
 \end{aligned}$$

A.3 Conclusion

Les grammaires catégorielles « reposent sur une conception strictement concaténative de l'assemblage des mots d'une phrase. » (Lecomte, 1996). En ce sens, elles ne peuvent pas rendre compte de certains phénomènes tels que les discontinuités et elles sont donc insuffisantes pour décrire les langues naturelles. Par contre, ce formalisme est tout à fait adapté pour une analyse partielle comme celle qui est pratiquée dans LOGUS ; il permet d'exprimer syntaxiquement et sémantiquement les dépendances locales centrées sur une unique tête lexicale.

Annexe B

λ Prolog

LOGUS est implémenté en λ Prolog ; le système pourrait bien sûr être réécrit dans un autre langage. Cependant, « *un langage de programmation peut plus ou moins donner prise à l'application de méthodes ou même induire ses propres méthodes* » (Ridoux, 1998) : λ Prolog est un langage de programmation assez particulier pour que son utilisation ait fortement influencé certains choix opérés.

Cette annexe veut présenter brièvement et aussi simplement que possible le langage λ Prolog. En première approximation (partiellement fausse, comme la plupart des approximations), λ Prolog apparaît comme une extension de Prolog, au niveau des formules manipulées d'une part et au niveau des termes utilisés d'autre part.

B.1 Les formules de λ Prolog

En programmation logique, les programmes sont des formules logiques. Leur exécution correspond à la recherche d'une preuve.

B.1.1 Prolog et clauses de Horn

À l'origine, Prolog¹ est conçu comme un démonstrateur de la logique classique. Il repose sur les propriétés particulières des *clauses de Horn*, clauses ayant au plus un littéral positif.

Un programme correspond à un ensemble d'hypothèses ; les clauses correspondantes (de Horn) ont exactement un littéral positif.

Par exemple,

$$\{g(x,y) \vee \neg h(x) \vee \neg f(x,y), f(x,x), f(x,y) \vee \neg k(y), h(A), k(B)\}$$

où g , h , k et f sont des symboles de prédicat, x et y des symboles de variables (implicitement quantifiées universellement), A et B des symboles de constante.

Le programme Prolog \mathcal{P} correspond à la conjonction des formules du premier ordre :

1. Le terme Prolog désigne ici un langage abstrait de programmation logique, sans référence particulière à l'une de ses implémentations.

Formule logique	Traduction Prolog
$\forall x \forall y (h(x) \wedge f(x,y) \Rightarrow g(x,y))$	$g(X,Y) \quad :- \quad h(X), f(X,Y).$
$\forall x f(x,x)$	$f(X,X).$
$\forall x \forall y (k(y) \Rightarrow f(x,y))$	$f(X,Y) \quad :- \quad k(Y).$
$h(A)$	$h(a).$
$k(B)$	$k(b).$

En Prolog, les variables (identificateurs qui commencent par une majuscule) sont quantifiées universellement au niveau de chacune des clauses.

Un *but* est une clause de Horn sans littéral positif. La forme particulière des buts et des clauses du programme permet de mettre en œuvre une stratégie particulière pour la résolution de Robinson, orientée par les buts et désignée par le terme de résolution SLD. Cette recherche est correcte ; elle est complète lorsque l'arbre est fini. Elle fournit en résultat les substitutions qui permettent d'aboutir à la résolution.

Ainsi dans l'exemple donné le but $\neg g(X,Y)$ donne en résultat de la résolution ($\mathcal{P} \wedge \neg g(X,Y)$) les substitutions $\sigma_1 = (X \leftarrow a, Y \leftarrow a)$, $\sigma_2 = (X \leftarrow a, Y \leftarrow b)$ qui rendent cette formule contradictoire ; ces substitutions rendent donc sa négation ($\mathcal{P} \Rightarrow g(X,Y)$) vraie ; elles correspondent aux valeurs de (X,Y) pour lesquelles $g(X,Y)$ est une conséquence logique du programme \mathcal{P} .

B.1.2 Extension de la programmation logique

En 1987, Miller, Nadathur et Scedrov proposent une nouvelle définition de la programmation logique (Miller et al., 1987). Pour eux, le calcul d'une preuve appartient à la logique intuitionniste (mieux appropriée a priori que la logique classique puisque si l'objet de la logique classique est d'attribuer des valeurs de vérité aux formules, celle de la logique intuitionniste est précisément de construire des preuves). Un langage de programmation logique est un ensemble de formules et de règles de déductions possédant de bonnes propriétés telles que la complétude et la correction pour une certaine forme de preuves dites *preuves uniformes* (preuves du calcul des séquents telles que les parties droites des séquents conclusions d'une règle d'introduction à gauche sont atomiques).

Prolog satisfait à cette définition qui a le mérite d'apporter un éclairage nouveau sur la résolution SLD. Cependant, l'intérêt essentiel de cette définition est que d'autres ensembles de formules y satisfont, permettant la définition de nouveaux langages de programmation logique.

B.1.3 Formules héréditaires de Harrop

L'ensemble des *formules héréditaires de Harrop* (désignées par D) et celui des buts (désignés par B) sont définis par les grammaires suivantes, mutuellement récursives, où A désigne une formule atomique :

$$\begin{aligned} D &::= A \mid D \wedge D \mid B \Rightarrow A \mid \forall x D \\ B &::= A \mid B \wedge B \mid D \Rightarrow B \mid \forall x B \mid \exists x B \end{aligned}$$

Les formules ainsi définies satisfont à la définition d'un langage de programmation logique au sens donné par Miller, Nadathur et Scedrov : λ Prolog.

Par rapport à Prolog, la sémantique opérationnelle de λ Prolog introduit dans le corps des clauses l'implication, la quantification universelle, la quantification existentielle et la disjonction. Les règles des séquents correspondantes sont données dans la figure B.1.

Disjonction à droite :

$$\frac{P \vdash A}{P \vdash A \vee B} \quad \frac{P \vdash B}{P \vdash A \vee B}$$

Quantification existentielle à droite :

$$\frac{P \vdash B[x \leftarrow t]}{P \vdash \exists x.B}$$

où t est un terme quelconque

Implication à droite :

$$\frac{P \wedge D \vdash B}{P \vdash D \Rightarrow B}$$

Quantification universelle à droite :

$$\frac{P \vdash B[x \leftarrow c]}{P \vdash \forall x.B}$$

où c n'admet pas d'occurrence libre dans B ni P

FIG. B.1 – Les nouvelles règles de calcul des séquents de λ Prolog.

Disjonction à droite : elle n'induit pas de nouvelles possibilités par rapport à Prolog ou la disjonction à droite se traduit simplement par la conjonction de deux clauses $((P_1 \vee P_2) \Rightarrow Q) \equiv (P_1 \Rightarrow Q) \wedge (P_2 \Rightarrow Q)$

Quantification existentielle à droite : dans Prolog, les variables sont quantifiées universellement au niveau des clauses et donc existentiellement au niveau des buts $(\forall x (P(x) \Rightarrow Q) \equiv (\exists x P(x) \Rightarrow Q))$. Donc, sur ce point encore, il n'y a pas de nouveauté par rapport à Prolog.

Implication à droite : il s'agit là de l'implication intuitionniste : on n'a pas $(D \Rightarrow B \equiv \neg D \vee B)$ et on ne peut pas prouver $\neg D$ pour prouver $D \Rightarrow B$. D'après la règle des séquents, obtenir la preuve de $(D \Rightarrow B)$ consiste à ajouter l'hypothèse D au programme pour obtenir la preuve du but B . Cette implication permet par exemple d'ajouter dynamiquement des hypothèses, sans utiliser, comme en Prolog, des prédicats extra-logiques (*assert* ou *retract*).

Quantification universelle à droite : pour démontrer $(\forall x B(x))$, on démontre $B(c)$ où c est une nouvelle constante qui ne doit apparaître ni dans B , ni dans P . Il s'agit donc d'une démonstration de $B(x)$ qui doit être la même pour tous les x concernés et non de la démonstration de $B(x)$ pour chacune des valeurs possibles de la variable x (démonstration *intentionnelle*). Cette sémantique opérationnelle est complètement inconnue dans le langage Prolog.

B.2 Les termes de λ Prolog

Motivés par le besoin d'introduire des variables de prédicats pour résoudre des problèmes de démonstration automatique, Miller et Nadathur proposent également une extension des termes manipulés par Prolog (Miller and Nadathur, 1986). Ils utilisent les λ -termes simplement typés du λ -calcul munis des α , β et η -équivalences, déjà présentés dans l'annexe A, page 135. Si restreindre le λ -calcul au λ -calcul simplement typé lui fait perdre sa complétude calculatoire, en revanche, la β -convertibilité de deux termes typés est décidable et les λ -termes simplement typés sont normalisables et même fortement normalisables : on peut atteindre leur forme normale par n'importe quelle suite de réductions².

L'unification (détermination des substitutions qui, si lorsqu'elles existent, permettent de rendre deux termes égaux) est l'un des éléments déterminants de la résolution Prolog. En logique du premier ordre, lorsque deux formules atomiques sont unifiables, il existe un plus grand unificateur (MGU : Most General Unifier), c'est à dire une substitution plus générale que les autres, unique, au changement de noms des variables près ; de plus, un algorithme simple permet sa détermination.

L'unification des λ -termes simplement typés n'est que semi-décidable et il n'existe pas toujours de solution plus générale unique. Le semi-algorithme de Huet permet sa mise en œuvre : il peut produire zéro, une, deux, plusieurs ou une infinité de solutions et conséquence de l'indécidabilité, la recherche peut ne pas se terminer (Huet, 1975). Si le typage des λ -termes ne suffit pas à rendre décidable leur unification, il la rend néanmoins possible et il joue un rôle essentiel dans les résultats mêmes de cette unification.

2. Pour les définitions et théorèmes concernant ces notions, on peut par exemple se reporter à (Lallement, 1990).

B.3 La programmation en λ Prolog

Il est impossible de rendre compte en quelques lignes des particularités de la programmation en λ Prolog³ et du pouvoir expressif de ce langage (qui font tout son intérêt car la programmation en clauses de Horn suffit à assurer la complétude calculatoire); on peut citer en particulier des exemples d'application donnés par S. Coupet-Grimal et O. Ridoux pour le traitement automatique des langues (Coupet-Grimal and Ridoux, 1995). Nous nous contentons d'insister ici sur l'importance du typage et de donner un exemple de l'intérêt du langage pour la manipulation de formules.

B.3.1 Le typage

Les deux premières parties de cette présentation ont pu faire croire que λ Prolog était une double extension de Prolog, au niveau des formules et au niveau des termes (ces diverses extensions se complétant les unes les autres (Belleannée et al., 1999)). Cette idée simple ignore que le typage est un élément essentiel de λ Prolog. Il joue le rôle de « *spécification partielle des programmes* » (Ridoux, 1998) et permet de détecter un grand nombre d'erreurs à la compilation. Par contre, il rend impossible certaines pratiques courantes (et discutables) en Prolog telles que par exemple l'utilisation de listes hétérogènes sans autres précautions.

B.3.2 Un exemple

λ Prolog est particulièrement bien adapté à la manipulation de formules (et donc pour la métaprogrammation (Ridoux, 1998)): les λ -termes et les quantifications explicites des variables permettent d'exprimer précisément le statut des variables et leur portée.

On trouve dans Sterling et Shapiro (Sterling and Shapiro, 1986) un programme destiné à faire un calcul de dérivation. Le prédicat *derive* prend en arguments la valeur de la fonction, la valeur de la variable et la valeur de la fonction dérivée. On y trouve des clauses telles que :

```

derive(X,X,s(0)).
derive(F+G,X,Fprime+Gprime) :-
    derive(F,X,FPrime),
    derive(G,X,GPrime).
derive(F*G,X,Fprime*G+Gprime*F) :-
    derive(F,X,FPrime),
    derive(G,X,GPrime).

```

Sous sa séduisante simplicité, ce programme cache quelques problèmes: d'abord C. Belleannée, P. Brisset et O. Ridoux font remarquer que cette écriture permet à des buts tels que *derive*(72,72,s(0)) d'être prouvables (Belleannée et al., 1999). Ensuite, elle ne permet pas la composition des fonctions; Sterling et Shapiro eux-mêmes reconnaissent que la résolution de ce problème demanderait une réécriture des clauses d'une manière beaucoup moins naturelle.

En λ Prolog, l'écriture d'un tel programme passe par une déclaration de types qui contraint à spécifier la nature des fonctions que l'on veut dériver. On peut par exemple décider que les

3. L'implémentation de λ Prolog utilisée est celle de l'IRISA de Rennes1: Prolog/MALI (Brisset, 1992; Ridoux, 1991).

fonctions en question font correspondre un scalaire à un scalaire et que les scalaires en question possèdent une structure de corps avec deux opérations *plus* et *prod* dont les éléments neutres sont respectivement *zero* et *un*. Ce choix conduit alors aux déclarations suivantes (*kind* introduit un constructeur de type, ici d'arité 0, et *type* introduit un constructeur de termes) :

```
kind scalaire type.
type (zero,un) scalaire.
type (plus, prod)
  scalaire → scalaire → scalaire.
```

Le prédicat *derive* prend en argument une fonction scalaire et sa dérivée, chacune représentée par un λ -terme (le type prédéfini *o* correspond au type des valeurs de vérité) ; on obtient la déclaration et les clauses suivantes ($x \setminus (F x)$ est l'écriture en λ Prolog de $\lambda x.(F x)$) :

```
type derive
  (scalaire → scalaire) → (scalaire → scalaire) → o.
derive x \ x \ un.
derive x \ (plus (F x) (G x)) x \ (plus (FPrime x) (GPrime x)) :-
  derive F FPrime,
  derive G GPrime.
derive x \ (prod (F x) (G x))
  x \ (plus (prod (FPrime x) (G x)) (prod (GPrime x) (F x))) :-
  derive F FPrime,
  derive G GPrime.
```

B.4 Conclusion

L'exemple donné est bien insuffisant pour montrer l'expressivité du langage. Il illustre néanmoins l'intérêt que peut présenter la programmation logique liée à l'usage des λ -termes.

Dans LOGUS, ces possibilités sont particulièrement exploitées lors de la segmentation en *chunks* où les β -réductions liées aux compositions des λ -termes sont directement assurées par le langage de programmation (cf. page 93). Elles le sont également pour la prise en compte du contexte (dépendances entre propositions et utilisation du contexte de dialogue, cf. page 111) ; l'implication intuitionniste permet en effet un ajout dynamique des hypothèses liées au contexte pour l'analyse de la suite du dialogue.

λ Prolog est un outil puissant qui demande un certain apprentissage. En particulier, les difficultés liées aux problèmes de l'unification peuvent tendre des pièges au programmeur débutant. Malgré sa richesse, il est relativement peu utilisé car il souffre de l'absence d'un manuel de programmation élémentaire, présentant quelques-unes de ses applications possibles.

Bibliographie

- Abeillé, A. and Blache, P. (2000). *Ingénierie des langues*, chapter Grammaires et analyseurs syntaxiques, pages 51–76. Hermès. cité page 35
- Abeillé, A. and Godard, D. (2000). French word Order and lexical Weight. In Borsley, R., editor, *The nature and function of Syntactic Categories, Syntax and Semantics*, volume 32, pages 325–358. Academic Press, New-York. cité page 56
- Abney, S. (1991). Parsing by Chunks. In *Principle Based Parsing*. R. Berwick, S. Abney and C. Tenny, Eds., Kluwer Academic Publishers. cité page 78, 83
- Abney, S. (1996). Partial parsing via finite-state cascades. In Carroll, J., editor, *Workshop on Robust Parsing ESSLLI'96*, pages 8–15. cité page 78
- Allen, J., Byron, D., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2000). An architecture for a generic dialogue shell. *Natural Language Engineering*, 6(3):1–16. cité page 15, 16
- Allen, J., Byron, D. K., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2001a). Towards Conversational Human-Computer Interaction. *AI Magazine*, 22(4):27–37. cité page v, 16, 17, 32, 34
- Allen, J., Ferguson, G., and Stent, A. (2001b). An Architecture for more Realistic Conversational Systems. In *Proceedings of Intelligent User Interfaces*, pages 1–8, Santa-Fe, NM. cité page v, 18
- Allen, J., Miller, B., Ringger, E., and Sikorski, T. (1996). Robust Understanding in a Dialogue System. In *34th annual Meeting of ACL*, pages 62–70. cité page 15
- Antoine, J.-Y., Bousquet-Vernhettes, C., Goulian, J., Kurdi, M.-Z., Rosset, S., Vigouroux, N., and Villaneau, J. (2002a). Predictive and Objective Evaluation of Speech Understanding: the "challenge" evaluation campaign of the I3 speech workgroup of the french CNRS. In *Proceedings of the LREC 2002, 3rd International Conference on Language Resources and Evaluation, Las Palmas, Spain*. cité page 22, 23, 81
- Antoine, J.-Y. and Caelen, J. (1999). Pour une évaluation objective, prédictive et générique de la compréhension en CHM orale : le paradigme DCR (demande, contrôle, résultat). *Langues*, 2:130–139. cité page 22
- Antoine, J.-Y. and Goulian, J. (2001). Étude des phénomènes d'extraction en français parlé sur deux corpus de dialogue oral finalisé. *TAL*, 42(2):413–440. cité page 57
- Antoine, J.-Y., Letellier-Zarshenas, S., Nicolas, P., Schadle, I., and Caelen, J. (2002b). Corpus OTG et ECOLE_MASSY : vers la collection de corpus francophones de dialogue oral diffusés librement. In *TALN'2002*, volume 1, pages 319–324, Nancy, France. cité page 52

- Antoine, J.-Y., Siroux, J., Caelen, J., Villaneau, J., Goulian, J., and Ahafhaf, M. (2000). Obtaining Predictive Results with an Objective Evaluation of Spoken Dialogue Systems : Experiments with the DCR Assessment Paradigm. In *LREC'2000, Athènes*. cité page 115
- Aït-Mokhtar, S., Chanod, J.-P., and Roux, C. (2002). Robustness beyond Shallowness : Incremental Deep Parsing. *Natural Language Engineering*, 8:121–144. cité page 77
- Aust, H., Oerder, M., Seide, F., and Steinbiss, V. (1995). The Philips automatic train timetable Information System. In *Speech Communication*, volume 17, pages 249–262. cité page v, 11, 12
- Austin, J.-L. (1962). *How to do things with words*. Oxford. cité page 40
- Balpe, J.-P. (2002). *Le Cerveau, le Langage et le Sens*, chapter La programmation du sens, pages 343–356. Odile Jacob, Paris, Université de tous les Savoirs edition. cité page 79
- Bar-Hillel, Y. (1964). *Language and Information*, chapter On Categorial and Phrase Structure Grammars. Addison-Wesley, Reading. cité page 80, 133
- Bargelli, D. and Lambek, J. (2001). An algebraic Approach to French Sentence Structure. In *Logical Aspects of Computational Linguistics*, pages 62–78, Le Croisic. Springer. cité page 94
- Baylon, C. and Mignot, X. (1995). *Sémantique du langage*. Nathan. cité page 43
- Bazzi, I. and Glass, J. (2000). Modeling Out-of-Vocabulary Words for Robust Speech Recognition. In *6th International Conference on Spoken Language Proc.*, Beijing, China. cité page 10
- Bazzi, I. and Glass, J. (2001). Learning Units for Domain-Independent Out-of-Vocabulary Word Modelling. In *Proceedings of the European Conference on Speech Technology (Eurospeech 2001)*, Aalborg, Danemark. cité page 10
- Bear, J., Dowding, J., and Shriberg, E. (1992). Integrating multiple knowledge sources for detection and correction of repairs in Human-Computer dialogue. In *ACL'92*, pages 56–63. cité page 36
- Belleannée, C., Brisset, P., and Ridoux, O. (1999). A Pragmatic Reconstruction of λ Prolog. *Journal of Logic Programming*, 41:67–102. cité page 143, 143
- Bennacef, S. (1995). *Modélisation du dialogue oral Homme-Machine : mise en œuvre dans une application de demande d'informations*. PhD thesis, Université de Paris XI - Orsay. cité page
- Bickmore, T. and Cassell, J. (2003). *Social Dialogue with embodied Conversational Agents*. Kluwer Academic, New-York, J. van Kuppevelt, L. Dybkjaer, N. Bernsen edition. cité page 18
- Blache, P. and Christo, A. D. (2002). Variabilité et dépendances des composants linguistiques. In *TALN'02*, volume 1, pages 205–214, Nancy, France. cité page 17
- Blanche-Benveniste, C. (1990). *Le français parlé ; études grammaticales*. CNRS Editions, Paris. cité page 53, 53, 102
- Blanche-Benveniste, C. (2002). L'étude grammaticale des corpus de langue parlée en français. In *Journées linguistiques de corpus*, Lorient, France. cité page 35, 35, 51, 53, 55, 55, 56, 56, 56, 56, 56, 77, 102
- Blanche-Benveniste, C., Rouget, C., and Sabio, F., editors (2002). *Choix de textes de français parlé*. Honoré Champion. cité page 53, 53, 53
- Bonneau-Maynard, H. and Devillers, L. (1998). Acquisition, transcription et annotation du Corpus Pariscorp. Technical report, Action de Recherche Concertée « Dialogue Oral » (Arc B2) de l'AUF. cité page 63

- Bonneau-Maynard, H., Devillers, L., and Rosset, S. (2000). Predictive performance of dialog systems. In *Int. Conf. on Language Resources and Evaluation Conference*, Athens. cité page 21
- Boualem, M. and Col. (2002). EURESCOM MUST project: multimodal, multilingual information services for small mobile terminals. In *TALN'02*, volume 2, pages 113–118, Nancy, France. cité page 18
- Boufaden, N., Delisle, S., Moulin, B., and Gouiaa, M. (1997). Linguistique informatique et robustesse : analyse syntaxique et sémantique pour la modélisation de dialogues finalisés. In *FRACTAL'97*, Grenoble, France. cité page 36
- Bousquet-Vernhettes, C. (2002). *Compréhension robuste de la parole spontanée dans le dialogue oral Homme-Machine - Décodage conceptuel stochastique*. PhD thesis, Institut de Recherche en Informatique de Toulouse, Toulouse, France. cité page 10, 30, 30, 31, 34
- Brisset, P. (1992). *Compilation de λ Prolog*. PhD thesis, Université de Rennes 1. cité page 143
- Bruce, B. (1975). Case Systems for Natural Language. *Artificial Intelligence*, 6:293–326. cité page 13
- Buszkowski, W. (2001). Lambek Grammars Based on Pregroups. In *Logical Aspects of Computational Linguistics*, pages 95–109, Le Croisic. Springer. cité page 94
- Chanet, C. (2001). 1700 occurrences de la particule *quoi* en français parlé contemporain : approche de la « distribution » et des fonctions du discours. *Marges linguistiques*, 2:56–80. cité page 55
- Christophe, A. (2002). *Le Cerveau, le Langage et le Sens*, chapter L'apprentissage du langage (les bases cérébrales du langage), pages 195–213. Odile Jacob, Paris, Université de tous les Savoirs edition. cité page 39
- Chung, G. (2000). Automatically incorporating unknown words in JUPITER. In *International Conference on Spoken Language Processing (ICSLP'00)*, volume 4, pages 520–523, Beijing, China. cité page 30
- Ciravegna, F. and Lavelli, A. (2002). Full Parsing for Information Extraction via finite-state Cascades. *Natural Language Engineering*, 8:145–165. cité page 77
- Colineau, N. (1997). *Étude des marqueurs discursifs dans le dialogue finalisé*. PhD thesis, Université Joseph Fourier, Grenoble, France. cité page 36, 54
- Coupet-Grimal, S. and Ridoux, O. (1995). On the Use of advanced Logic Programming Languages in Computational linguistics. *Journal of Logic Programming*, 24(1 et 2):121–159. cité page 143
- Danlos, L. (2002). *Le Cerveau, le Langage et le Sens*, chapter Linguistique informatique - traduction automatique, pages 261–270. Odile Jacob, Paris, Université de tous les Savoirs edition. cité page 1
- Danlos, L. (2003). Représentation sémantique sous-spécifiée pour les conjonctions de subordination. In *Actes de TALN*, volume 1, pages 95–104, Batz-sur-Mer, France. cité page 49
- Delsarte, P. and Thayse, A. (2001). *Logique pour le traitement de la langue naturelle*. Hermes Science Publications. cité page 76
- Dessalles, J.-L. (2000). *Aux origines du langage*. Hermes Sciences. cité page 32, 39
- Devillers, L., Maynard, H., and Paroubek, P. (2002). Méthodologies d'évaluation des systèmes de dialogue parlé : réflexions et expériences autour de la compréhension. *TAL*. cité page 20, 24

- Dzikovska, M., Allen, J., and Swift, M. (2003). Integrating linguistic and domain knowledge for spoken dialogue systems in multiple domains. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico. cité page 25
- Fillmore, C. J. (1968). *Universals in Linguistic Theory*, chapter The case for case, pages 1–90. Bach Emmon et Harms Robert. cité page 13
- Gadet, F. (1997). *Le Français ordinaire*. Colin, Paris. cité page 39, 52, 55
- Gazdar, G. (1993). *The Simulation of Human Intelligence*, chapter The handling of Natural Language, pages 150–177. Blackwell, Oxford, Donald Broadbent edition. cité page 5
- Girard, J. (1987). Linear Logic. *Theoretical Computer Science*, 50:1–102. cité page 135
- Glass, J. (1999). Challenges for Spoken Dialogue Systems. In *Proceedings IEEE ASRU Workshop*, Keystone, Colorado, USA. cité page 6, 34, 57
- Glass, J., Flammia, G., Goodine, D., Philips, M., Polifroni, J., Sakai, S., and Seneff, S. (1995). Multilingual Spoken-language Understanding in the MIT Voyager System. In *Speech Communications*, volume 17, 1-2, pages 1–19. cité page 15
- Goulian, J. (2002). *Stratégie d'analyse détaillée pour la compréhension automatique robuste de la parole*. PhD thesis, Université de Bretagne Sud. cité page 78
- Grice, H. (1979). Logique et conversation. *Communications*, 30:57–72. (1^{ère} Ed. New-York, 1975). cité page 43
- Hastie, H. W., Prasad, R., and Walker, M. (2002). What's the Trouble : Automatically Identifying Problematic Dialogues in DARPA Communicator Dialogue Systems. In *40th Annual Meeting of the Association of Computational Linguistics*. cité page 31, 31
- Hazen, T., Seneff, S., and Polifroni, J. (2002). Recognition Confidence Scoring and theirs Use in speech understanding systems. *Computer Speech and Language*, 16:49–67. cité page 10, 10
- Heeman, P. and Allen, J. (1999). Speech repairs, intentional phrases and discourse markers : modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, 25(4), 25(4):527–573. cité page 35
- Henry, S. (2002). Étude des répétitions en français parlé spontané pour les technologies de la parole. In *TALN'02*, pages 467–476, Nancy, France. cité page 52, 53
- Hirschman, L. (1998). Language understanding evaluations: lessons learned from MUC and ATIS. In *1st Int. Conf. Language Ressources and Evaluation, LREC'98*, pages 117–122, Grenad, Spain. cité page 22, 37
- Hirschman, L., Bates, M., Dahl, D., Fisher, W., Garofolo, J., Hunicke-Smith, K., Pallett, D., Pao, C., Price, P., and Rudnicky, A. (1992). Multi-site data collection for a spoken language corpus. In *Proc. DARPA Speech and Natural Language Workshop 92*, pages 7–14, Harriman, New York. cité page 22
- Holan, T., Kubon, V., Oliva, K., and Plátek, M. (1998). Two Useful Measures of Word Order Complexity. In *Coling-ACL's 98*, pages 21–28. cité page 57
- Holan, T., Kubon, V., Oliva, K., and Plátek, M. (2000). On complexity of word order. *TAL*, 41.1:273–300. cité page 57
- Huet, G. (1975). A unification algorithm for typed λ -calculus. *Theoretical Computer Science*, 1:27–57. cité page 142
- Jakobson, R. (1963). *Essais de linguistique générale*. Les Éditions de Minuit. cité page 42

- Kerbrat-Orecchioni, C. (1980). *L'énonciation*. Colin, Paris. cité page 43, 46
- Kerbrat-Orecchioni, C. (1986). *L'implicite*. Colin, Paris. cité page 43, 45
- Kerbrat-Orecchioni, C. (1990). *Les interactions verbales*, volume 1. Colin, Paris. cité page 17, 35, 41, 42, 45, 46, 51, 52, 56, 56, 102
- Kerbrat-Orecchioni, C. (2002). *Les actes de langage dans le discours : théorie et fonctionnement*. Nathan-Université. cité page 42
- Lallement, R. (1990). *Logique, réduction, résolution*. Masson, Paris. cité page 135, 142
- Lambek, J. (1958). The mathematics of sentence structure. *American mathematical monthly*, 65:154–169. cité page 134
- Lamel, L., Bennacef, S., Bonneau-Maynard, H., Rosset, S., and Gauvain, J.-L. (1995). Recent developments in spoken language systems for information retrieval. In *Proceedings of the European Conference on Speech Technology (Eurospeech'95)*, pages 207–210. cité page 12
- Lamel, L., Rosset, S., Gauvain, J.-L., Bennacef, S., Garnier-Rizet, M., and Proust, B. (2000). The LIMSI-ARISE System. In science B. V., E., editor, *Speech Communication*, volume 31, pages 339–353. cité page v, 13, 28, 29
- Lecomte, A. (1996). Grammaire et théorie de la preuve : une introduction. *Traitement Automatique des Langues (TAL)*, 37(2):1–38. cité page 137
- Lewin, I., Becket, R., Boye, J., Carter, D., Rayner, M., and Wirén, M. (1999). Language Processing for Spoken Dialogue Systems: is Shallow Parsing enough? In *ESCA ETRW Workshop on Accessing Information in Spoken Audio*, pages 37–42, Cambridge, UK. cité page 37
- Mariani, J., editor (2002). *Reconnaissance de la parole*. Hermès Sciences. cité page 8
- Martinie, B. (2001). Remarques sur la syntaxe des énoncés réparés en français parlé. *Recherches sur le français parlé*, 16:189–206. cité page 53, 53, 78
- Meng, H., Busayapongchai, S., Glass, J., Goddeau, D., Hetherington, L., Hurley, E., Pao, C., Polifroni, J., Seneff, S., and Zue, V. (1996). WHEELS : A Conversational System in the Automobile Classifieds Domain. In *International Conference on Spoken Language Processing (ICSLP'96)*, pages 542–545, Philadelphia. cité page 15
- Miller, D. and Nadathur, G. (1986). Some Uses of Higher-Order Logic in Computational Linguistics. In *24th Annual Meeting of the Association for Computational Linguistics*, pages 247–255. cité page 142
- Miller, D., Nadathur, G., and Scedrov, A. (1987). Hereditary Harrop formulas and uniform proof systems. In Gries, D., editor, *2nd Symp. Logic in Computer Science*, pages 98–105, Ithaca, NY. cité page 140
- Minker, W. (1996). Compréhension dans le domaine ATIS. In *Actes 21^{ièmes} Journées d'Étude sur la parole, JEP'96*, pages 417–420, Avignon, France. JEP'96. cité page 12, 75
- Minker, W. (1998a). Evaluation Methodologies for Interactive Speech Systems. In *LREC'98, Granada*, pages 199–206. cité page 22
- Minker, W. (1998b). Stochastic versus rule-based speech understanding for information retrieval. *Speech Communication*, 25:223–247. cité page 31
- Minker, W. (1999). *Compréhension automatique de la parole spontanée*. L'Harmattan. cité page 31, 31, 62, 75, 127

- Moeschler, J. (1995). L'interprétation des énoncés en contexte. In *5^{ième} École d'été de l'ARC, les rôles du contexte de de la situation dans la cognition*, volume C, pages 26–59. cité page 54
- Moeschler, J. (2000). *Introduction à la linguistique contemporaine*. Armand Colin. cité page 43, 48
- Montague, R. (1974). *Formal Philosophy*. Yale University Press, New Haven, USA. cité page 63
- Moulin, B. (1998). The Social Dimension of Interactions in Multiagent Systems. In *Lecture Notes in Artificial Intelligenc*, volume 144, Agents and Multi-Agents Systems, pages 109–122. Springer, W. Wobcke M. Pagnucco C. Zhang edition. cité page 37
- Oviatt, S. (2000). Multimodal Interface Research: A Science without borders. In *International Conference on Spoken Language Processing (ICSLP'2000)*, volume 3, pages 1–6, Beijing, China. Chinese Friendship Publisher's. cité page 18, 129
- Pallet, D. S., Fiscus, J. G., Fisher, W. M., Garofolo, J., Lund, B. A., Martin, A., and Przybocki, M. A. (1995). 1994 Benchmark Tests for the ARPA Spoken Language Program. In *Proceedings of ARPA Workshop on Spoken Language Technology*, pages 5–36. cité page v, 22
- Pekarek-Doehler, S. (2001). Dislocation à gauche et organisation interactionnelle. *Marges linguistiques*, 2:177–194. cité page 56, 56
- Pérennou, G. (1996). Compréhension du dialogue oral : le rôle du lexique dans l'approche par segments conceptuels. In *Actes de l'atelier Lexique et Communication Parlée, GDR-PRC Communication Homme-Machine*, pages 169–178. cité page 63
- Pierrel, J. and Romary, L. (2000). *Ingénierie des langues*, chapter Dialogue Homme-Machine, pages 331–349. Hermès. cité page 27, 29, 48
- Querler, N. L. (1996). *Typologie des modalités*. Presses universitaires de Caen. cité page 49
- Rabiner, L. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *IEEE*, 77(2):257–286. cité page 8
- Retoré, C. (1996). Calcul de Lambek et logique linéaire. *TAL*, 37(2):39–70. cité page 135
- Retoré, C. (2000). Systèmes déductifs et traitement des langues : un panorama des grammaires catégorielles. *Technique et Science Informatique, numéro spécial de Traitement automatique du langage naturel (TAL)*, 20(3):301–336. cité page 133
- Ridoux, O. (1991). *Maliv06 Tutorial and Reference Manual*. IRISA, Publication interne 611, Campus de Beaulieu, Rennes. cité page 143
- Ridoux, O. (1998). *λProlog de A à Z*. Université de Rennes 1, Rapport d'habilitation. cité page 139, 143, 143
- Rosset, S. (2000). *Stratégies et gestionnaire de dialogue pour des systèmes d'interrogation de bases de données à reconnaissance vocale*. PhD thesis, Université Paris XI, Orsay. cité page 20, 20, 46, 47, 49
- Rosset, S., Lame, L., Bennacef, S., Devillers, L., and J-L.Gauvain (1997). Corpus oral de renseignements touristiques. Technical report, AUPELF-UREF ACR-ILOR-B2. cité page 63
- Rousseau, D., Moulin, B., and Lapalme, G. (1996). Interpreting communicative acts and building a conversation model. *Natural Language Engineering*, 2(3):253–276. cité page 37
- Sabah, G. (1997). *Machine, langue et dialogue*, chapter Sens et traitement automatique des langues, pages 77–108. L'Harmattan. cité page 27, 39

- Savelli, M. (2001). *En fait en français contemporain : proposition d'analyse. Recherches sur le français parlé*, 16:63–82. cité page 55
- Searle, J. R. (1970). *Speech Acts*. Cambridge University Press. cité page 41
- Seneff, S. (1992a). Robust Parsing for Spoken Language Systems. In *Proc. ICASSP*, pages 189–192. cité page 15
- Seneff, S. (1992b). TINA: a natural language system for spoken language applications. *Computational Linguistics*, 18(1):61–86. cité page v, v, 14, 14, 14
- Seneff, S. (1998). The Use of Linguistic Hierarchies in Speech Understanding. In *International Conference on Spoken Language Processing*, Australie. cité page 14, 15, 25
- Seneff, S., Lau, R., and Polifroni, J. (1999). Organization, Communication, and Control in the GALAXY-II conversational System. In *Proceedings of the European Conference on Speech Technology (Eurospeech'99)*, pages 1271–1274, Budapest, Hongrie. cité page 15, 18
- Seneff, S. and Polifroni, J. (1996). A new restaurant Guide Conversational System: issues in Rapid Prototyping for Specialized Domains. In *International Conference on Spoken Language Processing (ICSLP'96)*, pages 665–668, Philadelphia. cité page 15
- Shriberg, E., Bates, R., Stolcke, A., and all (1998). Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech? *Language and Speech*, 41(3-4):439–487. Special Issue on Prosody and Conversation. cité page 18
- Shriberg, E. and Stolcke, A. (2002). Prosody Modeling for Automatic Speech Recognition and Understanding. In Ostendorf, M., Khudanpur, S., and Rosenfeld, R., editors, *Proc. of the Workshop of Natural Language Modeling*, Minneapolis. Institute for Mathematics and its Applications. cité page 18
- Shriberg, E. E. (1994). *Preliminaries to a Theory of Speech Disfluencies*. PhD thesis, Department of Psychology, University of California, Berkeley. cité page 52
- Siroux, J., Guyomard, M., Multon, F., and Rémondeau, C. (1997). Multimodal References in GEORAL TACTILE. In *ACL*, Madrid. cité page 18
- Sleator, D. and Temperley, D. (1991). Parsing English with a Link Grammar. Technical report, CMU-CS-91-196, CMU, USA. cité page 57
- Sowa, J. (2000). *Knowledge Representation*. Brooks/Cole Thomson Learning, USA. cité page 63, 73
- Sowa, J. F. (1984). *Conceptual Structures : Information Processing in Mind and Machine*. Addison-Wesley, Reading, MA. cité page 73
- Sperber, D. (2000). *Qu'est-ce que l'humain ?*, volume 2, chapter La communication et le sens, pages 119–128. Odile Jacob, Paris, Yves Michaud edition. cité page 1, 2, 43, 44
- Sperber, D. and Wilson, D. (1989). *La pertinence : Communication et cognition*. Éditions de Minuit. cité page 41, 44
- Stent, A. (2000). The Monroe Corpus. Technical Report 728, Department of Computer Science, University of Rochester. cité page 17
- Sterling, L. and Shapiro, E. (1986). *The Art of Prolog*. MIT Press. 1st edition. cité page 143
- van Benthem, J. (2000). *Applications of Mathematical Logic in Philosophy and Linguistics*, chapter Mathematical Logic and Natural Language : life at the border, pages 25–38. Kluwer. cité page 31

- Vanderveken, D. (1999). *Analyse et modélisation des discours. Des conversations humaines aux interactions entre agents logiciels*, chapter La structure logique des dialogues intelligents, pages 61–99. L'interdisciplinaire, B. Moulin edition. cité page 39, 41, 42
- Vanderveken, D. (2001). *Essays in Speech Act Theory*, chapter Universal Grammar and Speech Act Theory, pages 25–62. John Benjamin, Amsterdam Philadelphia, D. Vandervecken and Susumu Kubo edition. cité page 41, 42
- Vernant, D. (2003). Classification of speech acts revisited in a dialogical and actional perspective. In Caelen, J., Vanderveken, D., and Vernant, D., editors, *Logic and Dialogue*. Netherlands, Dordrecht, Kluwer. à paraître. cité page 41
- Villaneau, J., Antoine, J.-Y., and Ridoux, O. (2001). Combining Syntax and Pragmatic Knowledge for the Understanding of Spontaneous Spoken Sentences. In *Logical Aspects of Computational Linguistics*, pages 279–295, Le Croisic. Springer. cité page 81
- Walker, M., Litman, D., Kamm, A., and Abelia, A. (1997). PARADISE : A framework for Evaluating Spoken Dialogue Agents. In *Proceedings of the 35th Annual Meeting of Computational Linguistics, ACL97*. cité page 21
- Walker, M. A., Rambow, O. C., and Rogati, M. (2002). Training a Sentence Planner for Spoken Dialogue Using Boosting. *Computer Speech and Language*. cité page v, 21, 28, 28, 31
- Wilks, Y. and Catizone, R. (2000). *Encyclopedia of Microcomputers*, chapter Human-Computer Conversation. Dekker, New-York. cité page 5
- Young, S. (1999). Probabilistic methods in spoken dialogue systems. The Royal Society. cité page 31, 31
- Young, S. (2002). Talking to machines (statistically speaking). In *Proc. Int. Conf. Spoken Language*, Denver, Colorado. cité page 31, 31
- Zechner, K. (1998). Automatic Construction of Frame Representations for Spontaneous Speech in Unrestricted Domains. In *Proceedings of COLING/ACL 98, Montreal, Canada*, pages 1448–1452. cité page 78
- Zechner, K. (2001). Automatic Generation of Concise Summaries of Spoken Dialogues in Unrestricted Domains. In *24th ACM-SIGIR Int. Conf. Research and Development in Information Retrieval, New Orleans, LA*. cité page 78
- Zue, V., Seneff, S., Glass, J., Polifroni, J., Pao, C., Hazen, T., and Hetherington, L. (2000). JUPITER : Telephone-Based Conversational Interfac for Weather Information. *IEEE Transactions on Speeh and Audio Processing*, XX(Y):100–112. cité page 15
- Zue, V., Seneff, S., Polifroni, J., Philips, M., Pao, C., Goodine, D., Goddeau, D., Glass, J., and Brill, E. (1994). PEGASUS : A Spoken Language Interface for On-Line Air Travel Planning. In *ARPA Human Language Technology Workshop*, Merrill Lynch Conference Center, Princeton NJ. cité page 15

Index

- α -équivalence, 135
- β -équivalence, 135
- η -équivalence, 135
- λ -terme, 78, 135
 - simplement typé, 76, 135, 142
- abstraction, 72, 93, 135
- actes
 - assertifs, 40, 70
 - de langage, 40
 - déclaratifs, 41
 - directifs, 41, 69
 - expressifs, 41, 70
 - illocutoires, 40, 47, 69
 - illocutoires collectifs, 42, 47
 - locutoires, 40
 - perlocutoires, 40
 - promissifs, 41
- adj_num*, **87**
- adjectif*, **87**
- adjectif_pre_pose*, **87**
- adverbe*, **87**
- AMELIA, 28
- analyses
 - sélectives, 34, 76, 111, 124
 - syntaxiques partielles, **77**
- annexe*, **90**
- application, 135
- apprentissage, 8
- ARISE, 12, 28, 32
- ATIS, 12, voir évaluation, 75
- AT&T, 22, 28
- auxiliaire*, **90**
- bigramme, voir modèle bigramme
- black-box, voir boîte noire
- boîte
 - noire, 19, 115
 - transparente, 19
- c_adj*, **87**
- c_nom*, **87**
- c_nom_p*, **87**
- calcul
 - de Lambek, 78, 133
 - des séquents, 133, 137, 140
- CAP, **22**, 25, 48
- catégorie syntaxique, 85, **85**, 94, 99
- catlegs*, **86**
- chaîne d'objets, 64, **66**, 85
 - composée, **66**
 - simple, **66**
- chunk*, **77**, **83**, 93
 - minimal, **84**, 89, **97**, 102, 103, 123
- comparatif*, **87**
- compatible_famille_sem*, 101
- compétence linguistique, 44
- conj*, **86**
- conjonction*, **87**
- Cont_acte*, **71**, 73, **74**
- contenu expressif, **71**
- contenu propositionnel, **41**, 63, 92, 126
- coordination*, **90**, 106
- coordination entre objets, 64, **65**, 85
- Coord_obj*, **65**, **74**, 85
- corpus PARISCORP, **63**, 73, 111
- Curry-Howard (correspondance de), 134
- DCR, voir évaluation
- défi, voir évaluation
- demande*, **90**
- det*, **87**
- détermination
 - de propriété, 64, **65**, 99

- DOHM, 1, 5, 7, 25, 37, 56
 domaine, **27**
- enchaînement entre objets, 64, **65**
est_dependant_de, **99**
est_propriete_de(*EP*, *P*, *Et*), **99**
est_sous_obj_de, **97**
 étiquette
 d'objet, 64, **65**, 99
 de propriété, 64, **65**, 89, 99
Et_Obj, **65**, 72, **74**, 99
Et_Prop, **65**, 72, **74**, 99
 EURESCOM, 18
 évaluation
 ATIS, 21, 31, 115
 DCR, 22, 115
 MEDIA, 24, 126
 objective, 20
 par défi, 23, 75, 81, 115
 PEACE, 24, 115
 subjective, 20, 115
excuse, **90**, 106
expr, **87**
- F_acte*, **71**, 73, **74**
 factorisation (d'une chaîne d'objets), 68
 filtrage, **96**, 97, 123
 force illocutoire, **41**, 63, 92, 126
 forme
 indirecte, 70, 71
 partielle, 55
 totale, 55, 70
 forme développée, **67**
 forme expressive, **71**
- GALAXY-II, 15, 18
 GDR-I3, 23
 généricité, 86, 126
 GEORAL, 18
 glass-box, voir boîte transparente
 grammaires
 de Montague, 76, 134
 catégorielles, 78, 85, 93, 133
 de Montague, 133
 graphe conceptuel, 64, 73
- gv*, **87**
- Harrop (formules de), 140
hesitation, **90**, 106
 Horn (clauses de), 139
- illocutoire
 seeforce illocutoire ou logique illocutoire, 41
- implication
 intuitionniste, 111, 142
- implicatures, 43
 implicite, 43
 inférence, 43, 45
infinitif, **87**
infinitif_p, **87**
information, **90**
interjection, **87**
 interprétation
 contextuelle, 48, 92, **113**, 126
 littérale, 48
interregnum, **52**
interrogation, **90**
- JUPITER, 10, 15, 30, 34
- lexical (lexème), **84**
 LIMSI, 12, 28, 75
 logique
 illocutoire, 41, 63
 intuitionniste, 135, 140
 linéaire, 135
- LOGUS-I, **79**, 115, 116
 LOGUS-II, 82, **83**, 116
- magicien d'Oz, **23**, 56
 Markov, **8**
 MASK, 12
 maximes conversationnelles, 43
 MEDIA, voir évaluation, 115
meme_famille_sem, 101
 méthodes stochastiques, 30
 MMC, voir Markov
Mod_det_prop, **65**, **74**
 modèle
 acoustique, **8**

- bigramme, 8, 125
- du code, 42
- inférentiel, 43
- linguistique, 8, **8**
- N-gramme, 8, 32
- trigramme, 8, 125
- modèles de Markov, *voir* Markov
- modifieur de propriété, **65**
- Montague, 63, 76, 134
- mots
 - hors-vocabulaire, 10, 29, 90, 125
- multimodalité, 18
- MUST, 18
- N-gramme, *voir* modèle N-gramme
- neglige*, **90**
- neutre*, **90**, 102
- nomc*, **87**
- nomp*, **87**
- noyau principal, 110
- noyau sémantique, **107**, 118
- obj_acte*, **90**
- objet
 - minimal (d'une chaîne d'objets), **67**, 73
 - simple, 64, **66**
- OOV, 10, 29, 125
- Oz, *voir* magicien d'Oz
- PARADISE, 21
- PARISCORP, *voir* corpus PARISCORP
- particule*, **87**
- PEACE, *voir* évaluation
- PEGASUS, 15
- pertinence, 44, 49
- PHILIPS, 11
- pidgin, 32
- point d'interruption*, **52**
- pp*, **87**
- pragmatique, 40
- pre_c_nom*, **87**, 106
- pre_c_nom_p*, **87**, 106
- preobj*, **90**
- preposition*, **87**
- pronom*, **87**
- prop*, **90**
- propositionnel
 - seecontenu propositionnel, 41
- prosodie, 7, 11, 18
- r_inconnu*, **90**, 106
- REA, 18
- reconnaissance (erreurs de), 10, 29, 119–121, 125
- reconnaissance vocale, **7**, 121
- relation d'enchaînement, 64, 85
- relation d'objet à propriété, **65**, 100
- Rel_obj_Prop*, **65**, **74**, 100
- repair*, **53**
- reparandum*, **52**, 93, 123
- réparation, 35, **52**, 93, 100, 123
- répétitions, 52
- reprise*, **90**, 106
- requete*, **90**
- requête
 - fermée, 69, 70
 - ouverte, 69, 70
- rôle sémantique, 85, **86**, 94
- roles*, **89**, 90
- segment droit (d'une chaîne d'objets), 68, **68**, 111
- segmentation en
 - chunks*, 78, **78**, 80, 81, 83, **93**, 144
- sélectives, *voir* analyses sélectives
- Sem_Prop*, **65**, 72, **74**, 99
- sens utile*, **63**, 73, 76
- shallow parsing*, **77**
- solution optimale*, 96
- subjectivité, 46, 54
- superlatif*, **87**
- tâche, **27**
- TINA, **14**, 22, 30, 32, 34, 36, 76, 91, 111, 124
- traduction sémantique, 85, **90**
- travail de formulation*, **51**, 93
- trigramme, *voir* modèle trigramme
- TRIPS, 15, 32, 34
- unite*, 89, **90**

variabilité, 9, 55

verbe, **87**

VOYAGER, 15

WHEELS, 15