
LOGUS : compréhension de l'oral spontané

Présentation et évaluation des bases formelles de LOGUS

Jeanne Villaneau* — Olivier Ridoux** — Jean-Yves Antoine*

**Université de Bretagne Sud*
4 rue Jean Zay, F-56100 Lorient
{Jeanne.Villaneau, Jean-Yves.Antoine}@univ-ubs.fr

***IRISA - Université de Rennes 1*
Campus universitaire de Beaulieu, F-35042 Rennes cedex
Olivier.Ridoux@irisa.fr

RÉSUMÉ. Dans les systèmes de dialogue oral Homme-Machine, la compréhension de la parole spontanée est un problème difficile qui requiert des méthodes d'analyse robustes. La plupart des systèmes sont destinés à des actions très spécifiques : la compréhension repose sur la détection de mots ou segments clefs pour remplir les différents champs de requêtes prédéterminées. LOGUS, le système de compréhension présenté dans cet article s'appuie sur des formalismes logiques, grammaires catégorielles et graphes conceptuels, hors du champ habituel de leur utilisation. L'analyse, incrémentielle, construit une formule logique par compositions progressives des concepts reconnus de l'énoncé. L'article décrit et compare les deux premières versions du système. Leurs évaluations donnent des résultats prometteurs : elles font apparaître la bonne robustesse de l'analyse et son assez bonne capacité à reconstituer le sens des énoncés. La prise en compte plus large du contexte et la gestion du dialogue feront l'objet de travaux futurs.

ABSTRACT. In spoken dialogue systems, natural language understanding is a difficult problem for which robust parsing methods are required. Most of the systems achieve very specific tasks: understanding is founded on detection of key-words or patterns in order to identify values of pre-determined semantic frames. LOGUS, the understanding system we are presenting in this paper, uses logical formalisms, categorial grammars and conceptual graphs, outside their usual application field. The parsing is incremental; it builds a logical formula by gradually composing the recognized constituents of the sentence. The paper describes and compares the first two versions of LOGUS. Their evaluations yield promising results; they show the good robustness of the parsing and its quite good ability to reconstruct the meaning of the utterances. Future studies must be led in order to take into account the context more widely and to manage dialogue.

MOTS-CLÉS : compréhension automatique, parole spontanée, grammaire catégorielle, graphe conceptuel.

KEYWORDS: automated understanding, speech, categorial grammar, conceptual graph.

1. Introduction

La langue écrite a été longtemps l'objet essentiel des études sur la langue naturelle. Cependant, depuis quelques années, la langue orale et le dialogue prennent une place de plus en plus importante, tant dans le domaine de la linguistique que dans celui des traitements automatiques (TAL).

Progressivement en effet, l'étude des conversations « naturelles » est devenue l'une des priorités d'étude des linguistes. L'analyse du discours, la linguistique de l'énonciation et la théorie des actes de langage [AUS 62, SEA 70] ont fait émerger la notion de « *compétence communicative* » [HYM 72]. L'idée s'est alors imposée que la compétence communicative primaire est celle du dialogue oral.

En TAL, l'intérêt pour l'analyse des énoncés oraux est dû à des progrès récents dans le domaine de la reconnaissance de la parole. Ces progrès ont permis au Dialogue Oral Homme-Machine (DOHM par la suite) de devenir une réalité, comme en témoignent les serveurs vocaux mis à la disposition du grand public ces dernières années. En général, ces systèmes de DOHM accomplissent des tâches très spécifiques : horaires de train ou d'avion [LAM 00, MIN 99], informations météo [ZUE 00], etc. Élargir les applications et rendre le dialogue plus naturel posent des problèmes complexes, qui exigent des recherches interdisciplinaires, tant fondamentales que pragmatiques.

La figure 1 représente très schématiquement les différents composants que l'on trouve en général dans un système de DOHM. Le module de reconnaissance de la parole en est le premier maillon. À partir du signal sonore émis par le locuteur, il délivre une ou des listes de mots qui sont censés correspondre à l'énoncé source. En aval, le module de compréhension fournit au gestionnaire de dialogue une représentation sémantique de cet énoncé retranscrit. Ce dernier assure l'interface avec la base de données et propose des réponses ou éventuellement, des demandes d'information complémentaires. Le module de génération assure leur traduction dans la langue naturelle de l'utilisateur. La synthèse vocale permet leur retransmission sous la forme d'un signal sonore.

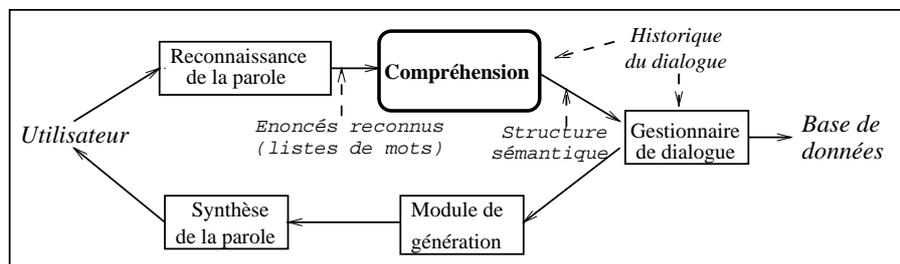


Figure 1. Système de DOHM

Le module de compréhension est donc un élément essentiel de ces systèmes ; or, sa tâche est difficile :

- les modules de reconnaissance de la parole délivrent une liste (ou un ensemble partiellement ordonné) de mots qui peuvent être erronés et entourent de « bruit » le message du locuteur¹ ;

- la forme des énoncés oraux présente des particularités inhérentes à leur mode de production : « *le travail d'élaboration est [donc] étalé en même temps que le produit élaboré* » [BLA 02]. Comme rien de ce qui a été dit ne peut être effacé, il contient les marques de la recherche des mots : hésitations, répétitions et auto-corrrections (désignés par les linguistes comme des « *réparations* »). De plus, dans la langue orale, le locuteur n'est pas tenu à une correction syntaxique aussi stricte que dans la langue écrite et les changements de parcours syntaxiques, voire sémantiques, y sont assez fréquents [BLA 90, KER 80].

Les retranscriptions orthographiques des énoncés transmises par le module de reconnaissance se prêtent donc difficilement à des analyses syntaxiques profondes : les modules de compréhension doivent adopter d'autres stratégies pour en extraire le sens. Le cadre très restrictif imposé aux énoncés par certaines applications permet par exemple l'utilisation de méthodes stochastiques pour générer des règles syntactico-sémantiques [MIN 96, ALL 96, SEN 92]. De plus, guidée par la tâche et grâce à une ambiguïté sémantique réduite, la compréhension peut aussi réduire à un strict minimum l'utilisation de la syntaxe ; ainsi, les analyses dites sélectives reposent sur la définition de cadres sémantiques de requêtes prédéterminées. Le repérage de certains syntagmes permet d'abord de déterminer à laquelle de ces requêtes préformatées correspond l'énoncé. Dans un second temps, il s'agit d'y détecter les segments conceptuels qui permettent de remplir les attributs de cette requête [MIN 96]. Le « dialogue » qui s'ensuit éventuellement consiste essentiellement à compléter les éléments manquants. La figure 2 donne un exemple d'analyse d'un énoncé par le système MASK (horaires de train) [LAM 95]. Les mots en gras sont les mots utilisés pour l'analyse. Les mots soulignés sont ceux qui permettent de détecter la nature des attributs.

De telles méthodes exploitent le caractère très étroit du domaine et de la tâche. Bien qu'elles aient prouvé leur efficacité pour des applications extrêmement finalisées, la question n'est pas tranchée de savoir si elles restent encore efficaces, ou même simplement utilisables, pour des tâches moins restrictives [HIR 98]. Il n'est pas certain non plus qu'elles permettent une compréhension assez fine des intentions du locuteur pour la mise en œuvre d'un dialogue plus coopératif [PIE 00].

Ces interrogations encouragent des tentatives de compréhension des énoncés oraux par d'autres voies. Les erreurs de reconnaissance et les spécificités de la parole orale spontanée imposent de toute façon l'utilisation de traitements robustes. Cependant, pour laisser davantage d'initiatives aux utilisateurs et mieux détecter leurs intentions, il convient que les structures sémantiques utilisées ne soient pas figées. L'analyse doit

1. Les taux d'erreurs de la reconnaissance de mot varient entre 5 et 25 % suivant la taille du vocabulaire et les conditions d'enregistrement du signal vocal [GAU 02].

être capable d'extraire de l'énoncé lui-même les relations entre les différents éléments qui le composent, plutôt que de les déduire par coïncidence avec une structure sémantique prédéfinie.

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

Figure 2. Requête et schéma sémantique pour l'analyseur MASK

LOGUS², le module de compréhension présenté dans cet article s'inscrit dans ce type de tentatives. L'analyse combine les approches syntaxique et sémantique : les règles utilisées s'appuient à la fois sur la nature syntaxique des éléments et sur une connaissance sémantique du domaine de l'application, définie par un certain nombre de prédicats. La représentation sémantique de l'énoncé est une formule logique à la Montague [MON 74] avec un formalisme simplifié, obtenue par composition.

Le domaine d'application choisi est celui du renseignement touristique, donc assez restreint pour que puisse être mise en œuvre une connaissance sémantique spécifique mais néanmoins assez étendu pour qu'il puisse sembler pertinent d'offrir une représentation sémantique de l'énoncé qui ne se réduise pas à un cadre préformaté.

La section 2 tente de spécifier précisément le problème posé et d'énoncer quelques principes généraux pour sa résolution. Elle montre aussi comment l'étude du fonctionnement de la première version du système, LOGUS-I, a entraîné une partielle remise en cause de ces principes. Les sections 3, 4 et 5 décrivent le formalisme utilisé et le fonctionnement du système sous sa forme actuelle, dans LOGUS-II. La section 6 est consacrée à l'analyse des résultats d'une évaluation comparée de LOGUS-I et de LOGUS-II. Nous essayons d'y décrire les points forts et les points faibles de l'approche. Enfin, dans la section 7, nous expliquons comment nous pensons compléter et améliorer le système.

2. LOGical Understanding System.

2. Principes généraux et première mise en œuvre

2.1. Compréhension et communication

La compréhension automatique d'un énoncé consiste en sa traduction d'une langue naturelle vers une langue formelle. Si l'on passe outre les objections de certains philosophes sur la possibilité même d'une telle traduction, le problème demeure de la richesse de la langue formelle utilisée. La logique est souvent choisie comme langue cible et des logiques non classiques telles que la logique modale [BEN 02] ont été construites pour améliorer la représentation du sens des énoncés de la langue naturelle. Delsarte et Thayse par exemple, proposent ainsi, pour le français, une interface langue naturelle/langue logique destinée à l'interrogation d'une base de données [DEL 01]. Cette interface concerne uniquement la langue écrite et aucune forme de dialogue n'est envisagée. Les énoncés traduits sont donc uniquement des demandes d'information. La « compréhension » est basée sur une analyse syntaxique complète des énoncés. Elle suppose l'existence en soi discutable d'un homomorphisme (appelé *homomorphisme syntagmatique* par les auteurs) de la syntaxe naturelle vers la syntaxe formelle. De plus, cette forme de communication répond à ce que les linguistes désignent par *modèle du code* : selon ce modèle, la langue naturelle permet au locuteur d'encoder le sens qu'il veut communiquer. Le récepteur décode l'expression qu'il reçoit et identifie le sens voulu.

Or, dès qu'il y a ébauche de dialogue, le modèle du code montre ses insuffisances. La fonction essentielle du langage est de servir la communication entre deux ou plusieurs partenaires. Même dans les situations de dialogue les plus simples, le sens linguistique de l'énoncé est souvent différent du sens voulu par le locuteur. Selon le *modèle inférentiel* développé par les linguistes, le récepteur doit *inférer*³ le sens voulu par le locuteur en tenant compte du contexte et de la pertinence de l'énoncé de son interlocuteur dans ce contexte (interprétation *pragmatique* [GRI 89]). Le contexte inclut des connaissances statiques diverses, par exemple générales et culturelles, et des connaissances dynamiques, telles que l'historique du dialogue et mêmes des « *suppositions sur l'état mental du locuteur* » [SPE 89]. Ce modèle prend également en compte le fait que la compréhension inférée est très souvent erronée ou incomplète : les « *ratés* » d'un dialogue en sont une partie intégrante. Par conséquent, élaborer un système de DOHM performant passe par la modélisation de différents contextes d'une part, et par la gestion des échecs partiels de la communication d'autre part. On conçoit alors les difficultés à vaincre pour y parvenir.

L'interrogation d'une base de données offre un cadre applicatif qui permet de simplifier bon nombre de paramètres. On peut en effet définir des connaissances sur les objets du domaine concerné et l'intention première du locuteur est supposée correspondre à une demande d'information (dans le domaine qui nous concerne, nous y ajoutons des demandes de réservation ou de location) : une grande partie du contexte

3. Selon Sperber [SPE 00], l'*inférence* diffère du raisonnement en ce qu'elle fait intervenir des processus inconscients.

est donc assez facilement définissable. Cependant, pour assurer un dialogue véritablement coopératif, il convient de pouvoir rendre compte d'autres types d'énoncés que de simples questions et requêtes ; aussi simple que puisse être la finalité du dialogue, le locuteur peut chercher à exprimer des actes de langage variés tels que informations, refus, confirmations, remerciements, etc. Dans l'état actuel du développement de ce travail, nous supposons qu'en aval de notre système, le module de dialogue gère les défauts et les erreurs de la communication. Cependant, ce sont les informations fournies par le module de compréhension qui rendent cette gestion possible.

2.2. Comment analyser l'oral spontané

Le calcul de la sémantique de Montague d'un énoncé a pour point de départ son analyse syntaxique, la représentation sémantique s'obtenant par traduction des relations syntaxiques obtenues. Or, comme il a été dit, l'analyse syntaxique des retranscriptions orthographiques transmises par la reconnaissance de la parole spontanée dans un système de compréhension automatique de la parole (CAP) semble illusoire.

2.2.1. Analyse partielle robuste : *chunking*

Depuis quelques années, des méthodes d'analyses partielles robustes ont été développées en TALN pour les textes écrits. Elles sont connues en anglais sous le nom de *shallow parsing*. Elles sont en particulier utilisées pour automatiser l'extraction d'informations des immenses quantités de données que représentent les textes électroniques [AiT 02]. Le principe général mis en œuvre dans ce type d'analyse consiste à produire des structures syntaxiques minimales, en général non récursives, désignées suivant différents termes : *clusters*, *chunks*, *segments* ou *chains*. Ces structures minimales sont constituées d'une tête à laquelle sont rattachés des termes situés à proximité immédiate. Par ailleurs, Abney fait valoir qu'elles correspondent à une réalité linguistique et même prosodique [ABN 91]. Ces traitements cumulent les deux avantages suivants :

1) ils sont robustes pour les champs d'application visés par les concepteurs de ces traitements ; en effet, les liaisons qu'ils effectuent sont sans ambiguïté car elles sont de portées plus petites que les structures syntaxiques supposées respectées dans ces applications. Les problèmes posés par les rattachements multiples possibles sont repoussés à des phases ultérieures d'analyse, s'appuyant sur d'autres règles. L'intégration de ces techniques de *shallow parsing* à des grammaires de *deep parsing* fait actuellement l'objet de travaux importants [BAN 99, FRA 03, SIM 03] ;

2) ils sont efficaces : les méthodes utilisées, souvent incrémentielles, conduisent à des analyses rapides.

La question se pose alors d'étendre le champ d'application de ces traitements au traitement de la parole. Or, des arguments linguistiques plaident en faveur de la possibilité de pratiquer cette étape dite de *chunking* sur les énoncés de l'oral spontané. En effet, les études comme celles de Blanche-Benveniste sur le français parlé [BLA 90] attestent que les structures syntaxiques minimales locales sont en général assez bien

respectées dans les réparations. Par exemple, dans les hésitations et auto-corrrections, les syntagmes sont la plupart du temps repris en leur début, comme dans : « *pour la euh vers la station enfin euh vers la station de métro* ». Ainsi, ce processus préserve des structures minimales de groupes de mots syntaxiquement cohérentes.

Pour autant qu'on adopte une définition de *chunks* qui soient contenus dans la portée des structures locales préservées dans l'oral spontané, une étape de *chunking* permet alors de relier certains mots grammaticaux à leur tête lexicale et d'éliminer ceux d'entre eux qui correspondent à des ratures (voir plus d'explication en section 4.2). Par ailleurs, les éléments ainsi obtenus étant plus « signifiants », il est possible d'utiliser des arguments sémantiques dans la suite du déroulement de l'analyse (cf. section 5).

De plus, à cause de la possible mauvaise reconnaissance des mots, il faut adopter une définition de *chunk* délibérément beaucoup plus court encore que ce que la littérature prévoit pour l'analyse de l'écrit.

2.2.2. Liens entre *chunks*

La construction de la représentation sémantique de l'énoncé pose le problème de la combinaison des *chunks*. Or, si dans la plupart des langues, comme c'est le cas en anglais et en français par exemple, l'ordre des mots est relativement respecté à l'intérieur d'un *chunk*, par contre, même dans ces langues dites à ordre fixe, les ordres des *chunks* dans un énoncé sont soumis à une certaine variabilité [ANT 99]. De plus, la nature des compositions entre *chunks* juxtaposés est, elle aussi, beaucoup plus variable dans la langue orale que dans la langue écrite. Il semble donc difficile de ne s'appuyer que sur des considérations syntaxiques pour déterminer la nature des liens entre certains syntagmes et, en particulier, pour déterminer la portée des coordinations et celle des négations, auxquelles s'ajoutent, dans l'oral spontané, celles des reprises et auto-corrrections. On peut cependant penser que la détermination de ces liens entre *chunks* peut être étayée par les connaissances sémantiques du système, si toutefois l'identité sémantique de ces *chunks* est suffisante. Il va de soi que ces connaissances seront d'autant plus faciles à mettre en œuvre que le domaine de l'application permettra de restreindre l'ambiguïté.

Ces remarques incitent à adopter les principes généraux suivants pour aborder l'analyse des énoncés proposés.

- 1) Dans une première étape, opérer une segmentation en *chunks* de l'énoncé, basée sur des relations syntaxiques locales.
- 2) Interpréter les constituants jugés « signifiants » obtenus au terme de la première étape afin de transformer les éléments syntaxiques en éléments sémantiques. Utiliser la sémantique (définir une connaissance sémantique) pour composer entre eux ces éléments.

La trace des compositions de *chunks* (et des compositions de compositions) constitue l'analogue d'un arbre d'analyse, ou si on se place d'un point de vue logique, d'un arbre de preuve. Dans le cas de la logique de Lambek [LAM 58], Retoré a montré [RET 00] comment la sémantique de Montague peut être attachée non pas à des

règles syntaxiques mais aux arbres de preuves en utilisant l'isomorphisme de Curry-Howard. Nous pensons donc qu'il est possible d'attacher à une analyse par *chunks* la construction d'une sémantique du premier ordre. Cependant, nous n'en avons pas ressenti le besoin dans le champ applicatif envisagé et nous n'avons utilisé que la logique propositionnelle.

2.3. Représentation du sens de l'énoncé

La représentation sémantique de l'énoncé donnée par LOGUS est une formule logique à la Montague avec un formalisme simplifié, sans quantificateurs. La formule est un λ -terme construit de façon compositionnelle. Un énoncé bien formé et non elliptique est en général obtenu par application d'un prédicat qui représente la nature de l'acte de langage (ou du moins les indices sur cette nature présents dans l'énoncé) à un objet simple ou composé⁴.

2.3.1. Représentation des objets

Les représentations sémantiques des objets du domaine sont construites avec les éléments de trois ensembles :

1) $\mathcal{ET}\mathcal{O}$ est l'ensemble des *étiquettes des objets*. Dans l'application test de LOGUS, il contient environ une centaine d'éléments :

hotel, cinéma, chambre, réservation, location, manger, aller, etc.

2) $\mathcal{ET}\mathcal{P}$ est l'ensemble des étiquettes des propriétés. Il contient une trentaine d'éléments qui se veulent relativement **indépendants** du domaine de l'application : *tarif, lieu, taille, identité, etc.*

3) \mathcal{DP} est l'ensemble des déterminations des propriétés. Par exemple, *double* et *cher* sont des déterminations de propriétés liées respectivement aux étiquettes de propriétés *taille* et *tarif*.

Une propriété est un couple $\langle EtP, DP \rangle$ où EtP appartient à $\mathcal{ET}\mathcal{P}$ et DP appartient à \mathcal{DP} . Par exemple, $\langle identité, (nom "Louvre") \rangle$ est une propriété. On considère une telle paire comme une application du λ -calcul.

Un *objet simple* est un couple $\langle EtO, LP \rangle$ où EtO appartient à $\mathcal{ET}\mathcal{O}$ et LP est une liste de propriétés. Par exemple, « un hôtel deux étoiles pas trop cher » a pour représentation sémantique :

$$\langle hotel, [\langle tarif, (pas\ cher) \rangle, \langle nbetoiles, (entier\ 2) \rangle] \rangle$$

Pour représenter l'objet précédent sous forme d'un λ -terme simplement typé, on définit un type simple appelé *trad_sem*. Les éléments de \mathcal{DP} sont de ce type. Les éléments de $\mathcal{ET}\mathcal{P}$ sont du type $(trad_sem \rightarrow trad_sem)$ et ceux de $\mathcal{ET}\mathcal{O}$ du type $(list\ trad_sem \rightarrow trad_sem)$. L'objet précédent correspond alors au λ -terme :

4. Cependant, certaines propositions, même non elliptiques, peuvent ne pas comporter d'objets ; c'est le cas par exemple d'énoncés tels que « merci » ou « au revoir ».

(*hotel* [(*tarif* (*pas cher*)), (*nbetoiles* (*entier 2*))])

Les objets peuvent être liés entre eux par des connecteurs de coordination : *et*, *ou*, *et_pas*, etc. ou par le connecteur de subordination *de* pour former des *objets complexes*. Ces différents connecteurs sont du type (*trad_sem* → *trad_sem* → *trad_sem*).

Une *chaîne d'objets* est un objet simple ou complexe. La chaîne d'objets est dite *simple* si elle ne contient pas de connecteurs de coordination (donc seulement des connecteurs *de*). Dans le cas contraire, elle est dite *complexe*. L'*étiquette d'une chaîne simple* est l'étiquette de l'objet le plus à gauche de la chaîne. Cette construction est utilisée en particulier pour calculer la sémantique d'une phrase étant données les sémantiques de ses *chunks* (voir section 5).

La sémantique choisie pour les connecteurs permet que toute chaîne d'objets puisse être mise sous la forme d'une coordination de chaînes simples, dite *forme normale* de la chaîne⁵. Par exemple, la formule suivante est une représentation de l'objet qui correspond à l'expression :

« *les tarifs d'une chambre double à l'hôtel Caumartin et au Crillon* »

(*de* (*tarif* []) (*de* (*chambre* [(*taille double*)]
(*et* (*hotel* [(*identité* (*nom* "Caumartin"))])
(*hotel* [(*identité* (*nom* "Crillon"))])))

Une forme normale de cette chaîne est :

(*et* (*de* (*tarif* []) (*de* (*chambre* [(*taille double*)]
(*hotel* [(*identité* (*nom* "Caumartin"))])))
(*de* (*tarif* []) (*de* (*chambre* [(*taille double*)]
(*hotel* [(*identité* (*nom* "Crillon"))])))

2.3.2. Représentation des actes de langage

Les prédicats qui représentent la nature des actes de langage (ou du moins les indices de cette nature) sont également des couples dont le premier élément exprime la *forme*⁶ de l'énoncé : *interrogation*, *requête*, *information*, *demande*, etc. et le deuxième élément l'expression de cette forme. Par exemple, « *est-ce que je peux* » se traduit par < *interrogation*, *possibilité* >, « *réservez-moi* » correspond au prédicat < *demande*, *existence* > appliqué à l'objet (*réservez-moi* []) alors que « *je n'ai pas réservé* » se traduit par le prédicat < *information*, *absence* > (appliqué à l'objet (*réservez-moi* [])).

En tant que λ -termes simplement typés, le premier élément du couple est de type (*trad_sem* → (*trad_sem* → *trad_sem*)) et le second est simplement du type

5. Une chaîne admet une seule forme normale, à l'ordre des chaînes simples coordonnées près.
6. Nous choisissons le mot *forme* car il ne s'agit que d'un indice linguistique de la force illocutoire au sens où la définit D. Vanderveken [VAN 01]. Le contexte de dialogue est nécessaire pour préciser l'acte de langage et les intentions du locuteur.

$trad_sem$. Le couple est donc de type $(trad_sem \rightarrow trad_sem)$, ce qui est le type d'un prédicat.

2.3.3. Graphes conceptuels

Le formalisme choisi permet que la formule sémantique obtenue soit directement convertible en un graphe conceptuel à la Sowa [SOW 84, SOW 00].

Pour la représentation sous forme de graphe conceptuel des objets, les concepts correspondent aux étiquettes des objets et aux représentations sémantiques des propriétés (les éléments des ensembles $\mathcal{ET}\mathcal{O}$ et \mathcal{DP}). Les relations conceptuelles correspondent aux étiquettes des propriétés (les éléments de $\mathcal{ET}\mathcal{P}$) et aux différents connecteurs. La figure 3 représente la chaîne d'objets précédente sous forme normale transformée en un graphe conceptuel. On peut remarquer que les concepts (boîtes angulaires) et les relations (boîtes arrondies) alternent le long d'une chaîne ; ils définissent une polarité.

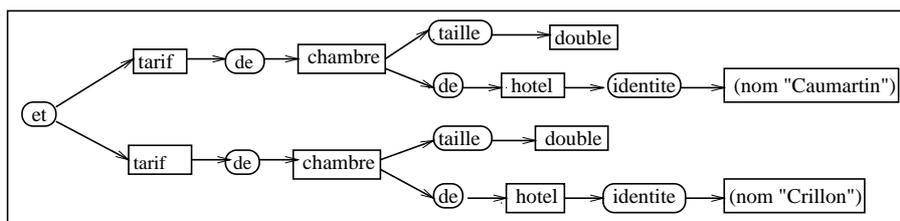


Figure 3. Graphe conceptuel représentant une chaîne d'objets

Pour ce qui est du couple qui représente l'acte de langage, son premier élément est considéré comme une relation conceptuelle et le second comme un concept.

2.3.4. Conclusion

L'un des objectifs essentiels est de parvenir à représenter le sens d'un énoncé sans faire appel à des cadres sémantiques prédéfinis. La représentation sémantique qui vient d'être présentée correspond à cette exigence. Les concepts correspondent à des briques élémentaires qui peuvent être rattachées entre elles pour fabriquer des représentations sémantiques partielles ou complètes, simples ou complexes. Cette représentation reste en même temps suffisamment simple pour pouvoir être facilement utilisable par le module de dialogue.

2.4. Première mise en œuvre des principes : LOGUS-I

L'architecture générale de LOGUS-I correspond à la mise en application des principes exposés à la fin de la section 2.2. Elle est schématisée dans la figure 4.

Quatre traitements successifs sont appliqués (les deux traitements principaux sont en gras dans la figure 4).

1) Chacun des lexèmes de la liste d'entrée présent dans le lexique est transformé en la liste de ses définitions (lemmatisation).

Une définition est composée d'une étiquette syntaxique et d'un λ -terme. 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 une tête lexicale, l'étiquette syntaxique est fractionnaire (au sens des grammaires catégorielles [BAR 64]) et le λ -terme est une abstraction. Par exemple, l'une des définitions du mot « *une* » est (*gn/nomc $\lambda x.(x (det indef sing))$*); *gn/nomc* est l'étiquette syntaxique d'un élément qui peut se combiner avec un nom commun situé à sa droite pour former un *chunk* nominal, le λ -terme correspondant détermine la composition avec ce nom commun suivant les règles qui vont être décrites.

2) Le *chunking* permet d'associer des mots juxtaposés sur la base de règles qui dépendent exclusivement de la syntaxe. Deux règles dérivées de celles des grammaires catégorielles de type *AB* sont appliquées aux étiquettes syntaxiques :

- (*A, A\B*) \rightarrow *B* : composition d'un élément de catégorie *A* avec un élément de catégorie fractionnaire *A\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 résultat s'obtient par composition des λ -termes correspondants comme dans l'exemple suivant où 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 :

$$\begin{array}{l} \text{« pas »} \quad (\text{adj/adj } \lambda x.(\text{pas } x)) \\ \text{« trop »} \quad (\text{adj/adj } \lambda x.x) \\ \text{« cher »} \quad (\text{adj cher}) \\ \hline (\text{adj } (\lambda x.(\text{pas } x)) (\lambda x.x \text{ cher})) \equiv_{\beta} (\text{pas cher}) \end{array}$$

3) Le troisième traitement est à la fois une interprétation et un filtrage. Les constituants non significatifs sont éliminés ; à ce stade, disparaissent ainsi les mots inconnus d'une part⁷ et les mots grammaticaux non rattachés d'autre part.

7. Dans un système de compréhension de l'oral spontané, les « *mots inconnus* » posent un problème incontournable et important. Les bribes de mots, les prononciations inhabituelles et les noms propres déformés font que le lexique de reconnaissance ne peut espérer couvrir tous les « mots » possibles. En général, ces mots dits hors-vocabulaire sont remplacés par un ou des mots phonétiquement proches et correspondent donc à des erreurs de reconnaissance. Cependant, l'impact de ces erreurs sur le comportement général du système est tel que certains systèmes font le choix de tenter de détecter ces mots hors-vocabulaire lors de la reconnaissance vocale ; ces mots sont alors étiquetés « inconnus ». Faute de disposer de tels systèmes de re-

Les autres constituants sont « interprétés », 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. Par exemple, le constituant (*adj (pas cher)*) est interprété comme une propriété éventuelle d'un objet du domaine : (*prop (pas cher)*).

4) Le quatrième et dernier traitement consiste à lier entre eux les constituants sémantiquement interprétés pour obtenir la formule logique finale. Les règles utilisées sont exclusivement sémantiques ; elles utilisent la connaissance sémantique définie sur les objets du domaine, décrite par des prédicats logiques. Cette connaissance constitue une sorte de « catalogue » des liens possibles entre les objets et leurs propriétés. L'analyse cherche 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 [VIL 01].

2.5. Évaluation de LOGUS-I et conclusions

LOGUS-I a participé à une campagne d'évaluation dite « par défi » [ANT 02] dans le cadre du groupe de travail 5.1 « compréhension robuste » du GDR I3 du CNRS.

L'évaluation des systèmes de compréhension est un problème presque aussi complexe que celui de leur élaboration et particulièrement difficile lorsqu'ils sont intégrés dans un système de dialogue oral. Le groupe de travail s'est donné pour objectif l'étude de ces évaluations. Il cherche quelles méthodologies mettre en œuvre pour que, au-delà des comparaisons de pourcentages d'analyses réussies⁸, l'évaluation permette de porter un diagnostic sur les points forts et faibles des systèmes en fonction des approches adoptées⁹.

Ainsi, la campagne d'évaluation à laquelle a participé LOGUS-I n'avait pas pour but d'obtenir une comparaison chiffrée entre les systèmes des différents participants (ces systèmes portaient d'ailleurs sur des domaines d'application différents ou sur des tâches distinctes d'un même domaine). Elle voulait essentiellement permettre à chacun d'entre eux de mieux cerner les défauts et les qualités de son système et, plus particulièrement, d'étudier son comportement sur certains phénomènes spécifiques de l'expression orale. LOGUS-I a ainsi été soumis à 1 200 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,

connaissance, les « mots inconnus » dont il est question dans cet article désignent uniquement les mots rencontrés dans les listes données en entrée à LOGUS et qui ne figurent pas dans son lexique.

8. Cette comparaison est déjà en soi difficile car elle exige que les domaines d'application et les tâches soient strictement identiques, que les réponses attendues aient la même forme sémantique et que les autres modules du DOHM n'interviennent pas dans les résultats.

9. Pour plus de détails sur le protocole utilisé, on pourra se reporter à [ANT 02].

avec des points de vue très différents, ont été très révélateurs des comportements des systèmes testés.

Les résultats obtenus (les chiffres et quelques commentaires concernant les différentes séries d'énoncés tests sont donnés à la section 6) montrent une grande robustesse de LOGUS-I 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 du *chunking* 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, à condition toutefois de faire le choix de *chunks* plus courts que ceux habituellement utilisés dans ce type d'analyse. En effet, le *chunking* peut engendrer des erreurs lorsque des « réparations » ou ellipses s'opèrent à l'intérieur d'un *chunk*. Par exemple, la segmentation « [dans un deux] [ou] [dans un trois étoiles] » risque d'entraîner une perte d'information (si le groupe de mots [« dans un deux »] est négligé) ou la nécessité de définir des coordinations complexes (les deux éléments à coordonner sont de nature différente).

C'est la phase d'établissement de liens entre *chunks* qui reconnaît des structures de taille plus habituelle. L'avantage de notre démarche est que ces structures auront pu faire l'objet d'hésitations ou de réparation que le *chunking* habituel ne sait pas traiter.

– Par ailleurs, les résultats obtenus valident également la construction des chaînes d'objets, utilisée pour éviter les cadres sémantiques prédéterminés (objets munis d'attributs considérés comme des cases à remplir obligatoires ou subsidiaires) : la base de connaissance spécifique aux objets du domaine, qui inventorie les liens possibles entre ceux-ci, correspond à la définition d'un contexte particulier. Cette méthode présente en particulier l'avantage de permettre d'analyser correctement les objets complexes (et donc la plupart des requêtes multiples), ce que les analyses sélectives ne font que très difficilement.

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 incises (« *un hôtel pas trop cher qu'est-ce que je peux dire deux étoiles par exemple...* »). Cette même absence de syntaxe ne permet pas de distinguer les différentes parties d'un énoncé 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* » ;

– 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 7), sont éliminés à la fin du

chunking. Cette élimination pure et simple aussi précoce accentue le problème de la reconnaissance des faux-départs et des incises ;

– par ailleurs et surtout, ce prototype est trop dépendant de l'application étudiée. L'étape d'interprétation qui suit le *chunking* contient une connaissance sémantique mal circonscrite. 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.

3. LOGUS-II : principes et formalisme

3.1. Les principes généraux de LOGUS-II

Trois principes directeurs président à la conception de LOGUS-II, la deuxième version de LOGUS [VIL 02].

1) La notion de *chunk* doit être modifiée, afin d'éviter les erreurs précédemment décrites (cf. 2.5). Dans LOGUS-II, le *chunking* est limité à la construction de *chunks minimaux*, un *chunk minimal* contenant au plus un élément lexical. Cette définition conduit par exemple à la segmentation suivante : [« dans un deux »] [« ou »] [« dans un trois »] [« étoiles »]. Cette segmentation offre la possibilité de traitements beaucoup plus simples des réparations et des coordinations.

2) Le second principe consiste à se donner la possibilité d'utiliser des arguments syntaxiques dans l'étape des liens 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 principe est de rendre génériques les règles utilisées. Certes, 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. Néanmoins, il nous semble important de ne pas négliger pour autant la généralité et la réutilisabilité. LOGUS a surtout été conçu pour mettre en œuvre et expérimenter un formalisme et quelques principes généraux. Pour prouver l'intérêt de l'approche, le système se doit d'être efficace ; or, 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.

A priori, vouloir s'appuyer sur les connaissances sémantiques spécifiques au domaine et en même temps prétendre concevoir des règles indépendantes de ce domaine peut sembler paradoxal. Néanmoins, le formalisme adopté dans LOGUS-II semble prouver qu'il est possible de concilier une certaine généralité des règles et l'utilisation d'une connaissance sémantique spécifique.

Les différentes étapes des traitements opérés dans LOGUS-II sont schématisées dans la figure 4. Par rapport à LOGUS-I, on peut noter la disparition de l'étape d'interprétation et l'ajout d'une étape supplémentaire, en fin d'analyse, destinée à étudier

les éventuelles dépendances entre propositions. Les sections suivantes décrivent ces différents traitements.

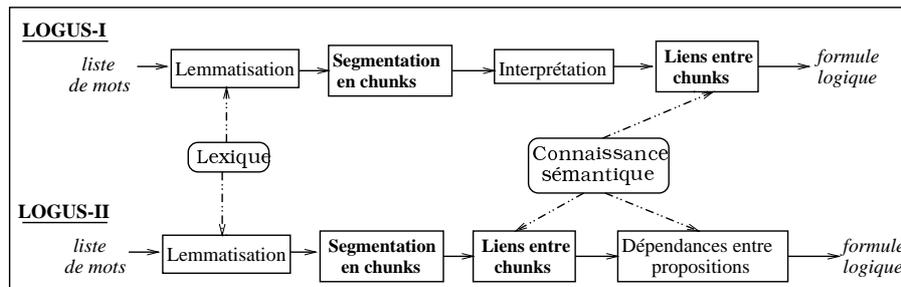


Figure 4. Architectures comparées de LOGUS-I et LOGUS-II

3.2. Modélisation des constituants

Le formalisme adopté dans LOGUS-II pour représenter les éléments de l'énoncé répond à plusieurs objectifs.

1) Le premier objectif est de pouvoir utiliser le même formalisme d'un bout à l'autre de l'analyse ; le *chunking* et les étapes suivantes ne diffèrent que par les règles de composition utilisées ; l'étape d'interprétation utilisée dans LOGUS-I, qui servait à passer d'un formalisme à l'autre devient donc inutile.

2) Le second objectif est d'offrir la possibilité d'associer, tout au long de l'analyse, les approches syntaxiques et sémantiques.

3) Enfin, il permet de définir des règles syntaxico-sémantiques indépendantes du domaine de l'application, suivant les principes qui vont être exposés.

Depuis les définitions du lexique jusqu'à la fin de l'analyse, les différents éléments de l'énoncé sont représentés par un triplet. De ce triplet, seul le troisième élément est strictement dépendant du domaine de l'application. Le principe général adopté pour préserver la généralité consiste à utiliser des règles dont la définition ne fait intervenir que les deux premiers éléments. Les règles peuvent donc être réutilisées dans un autre domaine sans être modifiées.

Les triplets en question sont de la forme $\langle Cat_synt, Role_sem, Rep_sem \rangle$ où :

1) Cat_synt est la *catégorie syntaxique*. Au niveau du lexique, cette catégorie peut être simple (par exemple, *adjectif*, *nomc* (nom commun), *nomp* (nom propre) sont des catégories syntaxiques simples) ou fractionnaire (au sens des grammaires catégorielles), les catégories fractionnaires étant éliminées en fin de *chunking* (voir détail dans la section 4.2).

D'autres catégories syntaxiques telles que *gn* (groupe nominal) et *gn_p* (groupe nominal prépositionnel) sont enrichies de traits qui sont déterminés lors de l'étape du *chunking*. Ainsi, *gn* et *gn_p* incluent dans leur définition les déterminants et prépositions qui servent à les définir. Par exemple, la catégorie syntaxique du *chunk* « vers l'hôtel » est

(*gn_p* (*gn nomc* (*det def sing*)) (*preposition ad*)).

Les étapes suivantes de combinaisons des *chunks* conduisent éventuellement à la construction de structures syntaxiques arborescentes, rendant ainsi possible la détermination syntaxique de chacun des éléments composés (voir par exemple la figure 4).

2) *Role_sem* correspond au rôle sémantique des éléments. Dans le lexique, ces rôles sémantiques peuvent être simples ou fractionnaires, avec, comme pour les catégories syntaxiques, élimination des rôles fractionnaires en fin de *chunking*.

Cet étiquetage sémantique correspond à une classification des différents constituants : par exemples, les objets ont pour rôle sémantique le rôle générique *objet* ; le rôle des propriétés est (*prop EtP*) où *EtP* correspond à l'étiquette de la propriété au sens défini précédemment, lorsqu'ont été exposés les éléments de construction de la représentation sémantique finale (cf. 2.3.1).

3) *Rep_sem* est la *représentation sémantique* de l'élément. Par exemple, pour un élément de rôle sémantique *objet*, elle correspond à la chaîne d'objet elle-même. Pour une propriété, il s'agit de la détermination de la propriété définie au 2.3.1.

Ainsi par exemple, $\langle \textit{infinitif}, \textit{objet}, (\textit{réservation []}) \rangle$ est la définition du mot « réserver » tandis que le lexème « y a-t-il » correspond au triplet : $\langle (\textit{gv 3 present}), \textit{interrogation}, \textit{il_y_a} \rangle$ où la catégorie syntaxique indique un groupe verbal à la troisième personne au présent¹⁰.

Les catégories syntaxiques sont totalement indépendantes de l'application et les rôles sémantiques le sont dans une très large mesure. Les règles étant définies à partir de ces deux éléments, elles sont génériques pour autant qu'ils le sont¹¹.

4. LOGUS-II : chunking

4.1. Les règles du chunking

Les deux premiers champs du triplet peuvent être de type fractionnaire (au sens des règles des grammaires catégorielles). La représentation sémantique correspondante est alors une abstraction au sens des λ -termes. Le regroupement des mots dans un *chunk* correspond à l'application des deux règles ci-dessous directement dérivées des

10. Ce lexique dépend de l'application dans le choix des mots et de leurs définitions : le mot « prix » ne fait référence qu'à la valeur vénale (et pas à la récompense) et le mot « étoile » ne se réfère qu'aux hôtels et restaurants.

11. La généralité des étiquettes des propriétés n'est pas totale, des propriétés telles que *lieu* et *temps* peuvent ne pas être des propriétés des objets du domaine. Les modifications à apporter aux règles restent néanmoins minimales.

deux règles des grammaires AB et étendues aux couples syntaxico-sémantiques de la catégorie syntaxique et du rôle sémantique.

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

et

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

La « représentation sémantique » du triplet issu de l'application de l'une des règles s'obtient par application à la représentation sémantique du triplet atomique de l'abstraction qui correspond à la représentation sémantique du triplet « fractionnaire ». Dans l'exemple suivant, *adj* correspond à adjectif, *c_adj* correspond à « chunk *adjectival* » et *prop* signifie *proprieté*¹² :

$$\begin{array}{l} \langle \text{« pas »} \rangle < c_adj/adj, ((prop R)/(prop R)), (\lambda x.(pas x)) \rangle \\ \langle \text{« trop »} \rangle < (adj/adj), ((prop R)/(prop R)), (\lambda x.x) \rangle \\ \langle \text{« cher »} \rangle < adj, (prop tarif), cher \rangle \\ \hline < c_adj, (prop tarif), (\lambda x.(pas x) (\lambda x.x cher)) \equiv_{\beta} (adj (pas cher)) \rangle \end{array}$$

La définition de deux relations d'ordre partiel, la première sur les catégories syntaxiques (\leq_C) et la seconde sur les rôles sémantiques (\leq_R), permettent la définition de deux nouvelles règles suivant les principes du calcul dans les prégroupes initiés par Lambek [BUS 01, BAR 01]. Dans ces règles, une forme de polymorphisme par sous-typage est obtenue en supposant que si une catégorie est demandée dans une composition, alors des catégories plus raffinées (c'est-à-dire des sous-types) sont également acceptées. Appliqué aux grammaires de type AB, ce principe conduit à la définition de la catégorie fractionnaire $(\forall x \leq A).(B/x)$ où x peut avoir des occurrences dans B . La nouvelle règle de calcul obtenue correspond à :

$$\text{si } C \leq A \text{ alors } ((\forall x \leq A).(B/x), C) \rightarrow B[x \leftarrow C]$$

où $B[x \leftarrow C]$ représente la catégorie obtenue en remplaçant les occurrences de x dans B par C . Comme dans les règles précédentes, ce calcul est appliqué aux deux premiers champs des triplets : la catégorie syntaxique et le rôle sémantique.

L'intérêt essentiel des deux nouvelles règles ainsi obtenues 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

12. La β -réduction est assurée par le langage de programmation : λ Prolog. Il s'agit d'un langage de programmation logique dont les termes sont les λ -termes simplement typés munis des α , β et η équivalences [MIL 86, BEL 99]. La version utilisée est Prolog/MALI, implémentée à l'IRISA de Rennes (France).

exemple, l'ordre $nom_c <_C (c_nom_ _) <_C (c_nom_p_ _)$ et la catégorie syntaxique $(\forall x \leq (c_nom_ _)) ((c_nom_p_ x (preposition _))/x)$ attribuée aux prépositions font que ces prépositions doivent pouvoir se composer avec un nom commun ou avec 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* ».

4.2. *Le chunking : la mise en œuvre*

Le lexique donne, pour chaque lexème connu, une liste de définitions dont chacune est un constituant (un triplet). Le *chunking* consiste à appliquer toutes les règles de compositions possibles entre les constituants jusqu'à épuisement de ces compositions. Une liste de constituants sur laquelle aucune règle ne s'applique plus est dite *solution du chunking*. Une *solution* est dite *optimale* si, parmi les solutions, elle est composée d'un nombre minimal de constituants. À l'issue du *chunking*, seules les *solutions optimales* sont retenues. Le caractère délibérément minimaliste des *chunks* (cf. 3.1) font que, dans la pratique, on obtient en général une seule *solution optimale*¹³.

Pour éviter une explosion combinatoire dans la recherche des *solutions optimales*, les énoncés sont préalablement fractionnés en sous-énoncés, les coupures intervenant entre deux lexèmes qui ne peuvent pas faire partie d'un même *chunk* (par exemple avant une préposition). Les solutions optimales sont alors obtenues par juxtaposition des solutions optimales de chaque fraction.

Enfin, le *chunking* se termine par l'élimination de certains constituants dans les solutions optimales (filtrage). En particulier, sont éliminés tous les constituants qui correspondent à des catégories syntaxiques et rôles sémantiques fractionnaires. Ainsi, le *chunking* constitue un premier traitement des hésitations et des reprises : dans l'exemple « *pour la euh vers la station enfin euh vers la station de métro* », le premier « *pour la* » est éliminé.

Les tableaux 1 et 2 donnent un exemple du *chunking* sur un énoncé complet. Pour simplifier ces tableaux, seule la définition qui sert à l'élaboration de la solution optimale est représentée, en face du lexème concerné. Les tableaux montrent 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 constituants. Le premier d'entre eux disparaît à l'issue du *chunking* et les 14 autres sont soumis aux règles syntaxico-sémantiques pour la suite de l'analyse qui est donnée dans les tableaux 3, 4, 5 et 6.

13. Une étude menée sur 100 des 1200 énoncés tests (cf. 2.5) ne montre aucun exemple de multiplicité des solutions optimales. En revanche, dans 61% de ces énoncés, un ou plusieurs des *chunks* obtenus admettent plusieurs définitions.

<i>je</i>	< (pronom pers 1 sing), annexe, neutre > ∅
<i>j'ai réservé</i>	< (gv 1 passe)/(pp "avoir"), < info, obj_acte > /obj_acte, λx.(présence x) > < (pp "avoir"), obj_acte, (réservation []) > < (gv 1 passe), < info, obj_acte >, (présence (réservation [])) >
<i>une chambre</i>	< (gn nomc (det indef sing))/nomc, obj_acte/obj_acte, λx.x > < nomc, obj_acte, (chambre []) > < (gn nomc (det indef sing)), obj_acte, (chambre []) >
<i>dans un deux</i>	< (∀x ≤ (pre_gn _)).(pre_gn_p x (prep. in))/x, R/R, λx.x > < (∀x ≤ adj.).(pre_gn x (det indef sing))/x, (prop A)/(prop A), λx.x > < (adj_num inf_10), (prop int), (ent. 2) > < (pre_gn_p (pre_gn (adj_num inf10) (det indef sing)) (prep. in)) (prop int), (entier 2) >
<i>étoiles</i>	< nomc, (unite objet), λx (hotel [(nbetoiles x)]) > < nomc, (unite obj_acte), λx (hotel [(nbetoiles x)]) >
<i>à l'hôtel</i>	< (∀x ≤ (gn _)).(gn_p x (prep. a_prep))/x, R/R, λx.x > < (gn nomc (det def sing))/nomc, obj_acte/obj_acte, λx.x > < nomc, obj_acte, (hotel []) > < (gn_p (gn nomc (det def sing)) (prep. a_prep)), obj_acte, (hotel []) >
<i>euh</i>	< particule, hesitation, neutre > < particule, hesitation, neutre >
<i>à l'hôtel</i>	< (∀x ≤ (gn _)).(gn_p x (prep. a_prep))/x, R/R, λx.x > < (gn nomc (det def sing))/nomc, obj_acte/obj_acte, λx.x > < nomc, obj_acte, (hotel []) > < (gn_p (gn nomc (det def sing)) (prep. a_prep)), obj_acte, (hotel []) >
<i>Caumartin</i>	< nomp, (prop identité), (nom "Caumartin") > < (nomp, (prop identité), (nom "Caumartin")) >
<i>pour y aller</i>	< (infinitif_p (prep. pour))/infinitif, obj_acte/obj_acte, λx.x > < infinitif, obj_acte, (aller [(vers obj_contexte)] >, > < (infinitif_p (prep. pour)), obj_acte, (aller [(vers obj_contexte)]) >
<i>d'ici</i>	< (∀x ≤ (gn _)).(gn_p x (prep. ab))/x, obj_acte/obj_acte, λx.x > < (pronom lieu 3 sing), obj_acte, lieu_contexte > < (gn_p (pronom lieu 3 sing) (prep. ab)), obj_acte, lieu_contexte >

Tableau 1. Exemple de segmentation en chunks sur un énoncé complet (suite et fin page suivante)

<i>comment</i>	< <i>adverbe, interrogation, comment</i> >
	< adverbe, interrogation, comment >
<i>est-ce-que</i>	< <i>expr, interrogation, oui_non</i> >
	< expr, interrogation, oui_non >
<i>je</i>	< <i>(gv 1 B)/(verbe _ B), R/R, λx.x</i> >
<i>peux</i>	< <i>(verb 1 pres.), modalite, pouvoir</i> >
	< (gv 1 pres.), modalite, pouvoir >
<i>faire</i>	< <i>infinitif, annexe, neutre</i> >
	< infinitif, annexe, neutre >

Tableau 2. Exemple de segmentation en chunks sur un énoncé complet (suite et fin)

5. LOGUS-II : connaissance sémantique - règles syntaxico-sémantiques

Les étapes qui suivent le *chunking* utilisent des règles syntaxico-sémantiques qui s'appuient sur une connaissance sémantique du domaine. Celle-ci constitue une base de données des liens qui peuvent être établis entre les objets du domaine et leurs propriétés, pour la formation de la formule finale. Les règles syntaxico-sémantiques permettent la mise en œuvre de ces liens.

Le formalisme choisi, tant pour la connaissance sémantique que pour les règles syntaxico-sémantiques, fait que, lorsqu'elles portent sur les objets de ce domaine ou sur leurs propriétés, ces règles ne dépendent que des relations définies entre les objets et non des objets eux-mêmes. Ainsi, elles peuvent être appliquées à tout autre ensemble d'objets et de propriétés sur lequel seraient définies les mêmes relations conceptuelles.

5.1. Les prédicats de la connaissance sémantique

La majeure partie de la connaissance sémantique est constituée de prédicats dont les arguments sont des objets du domaine de l'application. Elle en est donc directement dépendante. Il est cependant possible de faire le choix de prédicats qui soient eux-mêmes génériques, et donc réutilisables sur le système d'objets d'un autre domaine.

Les prédicats dits « de base » définissent la connaissance sémantique proprement dite. D'autres prédicats relèvent également de cette connaissance mais ils sont définis comme des conséquences des prédicats précédents.

5.1.1. Les prédicats « de base »

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

– *est_sous_obj_de* : ce prédicat prend pour arguments deux étiquettes d'objets. La relation *est_sous_obj_de(Et1, Et2)* est vraie si deux objets simples *O1* et *O2*, d'éti-

quettes respectives $Et1$ et $Et2$ peuvent être liés l'un à l'autre pour former la chaîne d'objets simple ($de O1 O2$), par exemple $est_sous_obj_de(chambre, hotel)$ permet de construire la chaîne simple ($de (chambre _) (hotel _)$) ou un graphe conceptuel élémentaire (cf. figure 5).

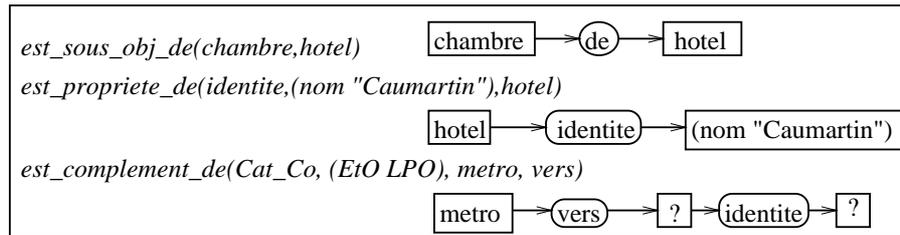


Figure 5. Exemples de graphes conceptuels élémentaires définis par la connaissance sémantique

– $est_propriete_de$: il s'agit d'un prédicat d'arité 3 dont les deux premiers arguments sont une étiquette de propriété et sa détermination, le troisième une étiquette d'objet. La relation $est_propriete_de(EtP, DP, EtO)$ correspond à la possibilité de rattacher la propriété ($EtP DP$) à un objet d'étiquette EtO : par exemple, $est_propriete_de(identite, (nom "Caumartin"), hotel)$ (cf. figure 5).

– $est_complement_de$: c'est le prédicat le plus complexe de cette connaissance sémantique. Il inventorie les possibilités pour un objet CO d'être considéré comme l'une des propriétés d'un autre objet O d'étiquette Et . La relation est conditionnée par la catégorie syntaxique Cat_CO de l'élément qui correspond à l'objet CO . Les quatre arguments du prédicat sont Cat_CO , CO , Et et $Lien$, où $Lien$ est la relation conceptuelle qui correspond au rattachement ($Lien$ joue donc le rôle d'une étiquette de propriété).

Par exemple, $est_complement_de(Cat_CO, (EtO LPO), metro, vers)$ si

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

Une telle relation correspond également à un graphe conceptuel élémentaire (cf. figure 5).

– est_dans : ce prédicat d'arité deux permet essentiellement des simplifications dans la construction des chaînes ; il prend en arguments deux étiquettes d'objet. Ainsi $est_dans(dormir, hotel)$ permet que les objets d'étiquette *dormir* et *hotel* soient fusionnés en un seul objet d'étiquette *hotel*, avec réunion des propriétés correspondantes. Ainsi, l'expression « *dormir ce soir dans un hôtel deux étoiles* » peut être considérée comme relative à un seul objet d'étiquette *hotel* avec les deux propriétés « *ce soir* » et « *deux étoiles* ».

5.1.2. *Les prédicats « conséquences »*

Des relations sémantiques peuvent être établies entre les objets ou entre leurs propriétés sur la base des prédicats précédents.

L'un des plus importants d'entre eux est le prédicat *comparables*, qui joue un rôle important comme indice de la portée des coordinations, reprises et négations. On veut déterminer si deux objets doivent en priorité se subordonner l'un à l'autre, ou s'ils peuvent occuper la même position dans le développement paradigmatique de l'énoncé. On adopte donc la définition suivante : deux objets simples *comparables* sont deux objets de catégories syntaxiques compatibles qui ne peuvent être subordonnés l'un à l'autre dans une chaîne.

5.2. *Règles syntaxico-sémantiques*

Les règles syntaxico-sémantiques ont pour but de déterminer les possibilités de compositions entre deux ou trois constituants consécutifs pour obtenir un constituant résultat.

Leur forme générale est $\langle L_Const, L_Cond, Const_Res, Reg_Res \rangle$ où :

- *L_Const* est la liste des constituants à composer,
- *L_Cond* est la liste des conditions que doivent vérifier ces constituants,
- *Const_Res* est le constituant résultat de l'application de la règle,
- *Reg_Res* sont les règles de construction de ce constituant résultat.

5.2.1. *Niveaux de règles*

Les études sur la langue parlée montrent le respect de structures syntaxiques locales ; c'est cette propriété qui est utilisée pour le *chunking*. Elles attestent 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) [KER 90, BLA 90]. Cependant, elles permettent de conclure également que les énoncés de l'oral spontané ne sont pas agrammaticaux et qu'il est « inutile d'envisager une syntaxe spécifique de l'oral » [BLA 02].

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 définir des priorités dans les compositions par des « niveaux » de règles.

Par une double application de cette règle, l'analyse de l'expression « *les tarifs pour une chambre à l'hôtel Caumartin* » aboutit à la construction de l'objet

(de (tarif []) (de (chambre []) (hotel [(identité (nom "Caumartin"))])))).

Les règles définies à ce niveau¹⁵ doivent donc permettre, en principe, de relier entre eux tous les syntagmes d'un énoncé lorsque celui-ci respecte les conditions de « corrections » syntaxique et sémantique imposées. Par exemple, la règle de coordination suivante entre deux objets vérifie la coïncidence d'indices syntaxiques (même prépositions) et sémantiques (objets *comparables*) :

$$\begin{array}{l}
 L_Const = \langle C1, obj., O1 \rangle, \langle -, (coord. "et"), - \rangle, \langle C2, obj., O2 \rangle \\
 L_Cond = \begin{cases} C1 \text{ et } C2 \text{ sans préposition ou avec la même préposition} \\ comparables_objets(O1, O2) \end{cases} \\
 \hline
 Const_Res = \langle (conj C1 C2 et), objet, (et O1 O2) \rangle \\
 Reg_Res = []
 \end{array}$$

5.2.1.3. Troisième niveau

Les règles du troisième niveau permettent d'opérer des compositions similaires à celles du second niveau, tout en augmentant la flexibilité : des variations sont autorisées dans l'ordre des constituants (modification de l'ordre des éléments de *L_Const*), ainsi que dans l'emploi des prépositions et des déterminants (simplification de *L_Cond*). De plus, ce niveau autorise le « saut » de certains constituants, par exemple, celui des mots inconnus, ce qui constitue un premier traitement, certes très élémentaire, de ces mots. Par exemple, la coordination indiquée par le « *et* » dans l'expression « *les tarifs dans un deux étoiles et un trois étoiles aussi* » sera opérée par une règle du troisième niveau, à cause de la non-répétition de la préposition « *dans* ». Le troisième niveau donne aussi une mesure formelle de la flexibilité requise pour produire chaque interprétation. Elle sera utilisée dans la suite du traitement.

5.2.2. 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 du *chunking*. 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.

Outre ses avantages « logiques », cette composition « en cascades » permet d'éviter une explosion combinatoire lors de l'application des règles. Si toutes les règles sont

15. Le second niveau comporte environ soixante-dix règles, ce nombre étant, comme le précédent, peu significatif en soi.

appliquées sur les constituants d'un énoncé comportant de multiples reprises et auto-corrections, le temps de traitement devient rapidement redhibitoire¹⁶.

5.2.3. Noyaux sémantiques

Les traitements précédents sont insuffisants pour les énoncés très disloqués. Un traitement supplémentaire est donc mis en œuvre lorsqu'à l'issue de l'application des règles, l'énoncé se présente sous la forme d'éléments juxtaposés, a priori incompatibles. Il s'agit de récupérer une partie du sens de l'énoncé.

L'hypothèse utilisée ici 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 sémantique. 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 sémantique). L'analyse peut n'être que partielle : le résultat correspond aux différentes associations opérées. Le traitement se déroule en deux étapes successives :

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 ;

2) ensuite, l'analyse tente de rattacher sémantiquement les autres éléments de l'énoncé à ces noyaux, en utilisant les prédicats de la connaissance sémantique (voir section 5.1). Un élément isolé qui ne peut se rattacher à l'un des noyaux est simplement ignoré. Ainsi, l'analyse permet d'extraire d'un énoncé apparemment disloqué des parties sémantiquement cohérentes, en prenant en compte le plus grand nombre possible de ses éléments.

Par exemple, à l'issue de l'application des trois premiers niveaux de règles, l'énoncé « *une chambre au Caumartin le prix c'est quoi pour une double* » correspond aux trois constituants [« *une chambre au Caumartin* »] [« *le prix c'est quoi* »] [« *pour une double* »]. Les deux premiers éléments sont des noyaux sémantiques ; cependant, comme le second porte la marque de l'interrogation et comme il est de plus un groupe verbal, il est considéré comme le « noyau principal ». La chaîne d'objet qui correspond au premier noyau : *(de (chambre [])) (hotel [(identité (nom "Caumartin"))])* se rattache sémantiquement au « noyau principal ». Ensuite, le dernier élément [« *pour une double* »]

16. Il ne doit pas s'écouler plus de quelques secondes entre une intervention du locuteur et la réponse du système. Le temps de traitement est donc une question cruciale à chaque étape.

peut s'insérer dans la chaîne en tant que propriété de l'objet d'étiquette *chambre*, pour obtenir la formule finale :

((*interrogation quel*) (*de* (*tarif* [])
(*de* (*chambre* [(*taille double*)])
(*hotel* [(*identité (nom "Caumartin")*)])]))))

5.2.4. Suite de l'analyse de l'énoncé-exemple

Les tableaux 3 et 4 indiquent les constituants résultants de l'exemple après application des règles des niveaux 1 et 2. Le niveau 1 permet la reconstitution des syntagmes nominaux complets « *dans un deux étoiles* » et « *à l'hôtel euh à l'hôtel Caumartin* ». Dans l'application des règles de niveau 2, la non-répétition de la préposition *dans* devant l'expression « *à l'hôtel* » fait que les deux objets correspondants ne sont pas composés. De même, la position de l'expression « *pour y aller d ici* » ne permet pas sa liaison avec « *comment est-ce que je peux faire* ». Après l'application des règles de niveaux 3 et 4, l'énoncé est composé de deux constituants dont les rôles sémantiques et les représentations sémantiques sont donnés dans le tableau 5.

je	< (<i>gv 1 passe</i>), < <i>info, obj.</i> >
j ai réservé	< <i>présence, (réservation, [])</i> >>
une chambre	< (<i>gn nomc (det indef sing)</i>), <i>obj.</i> , (<i>chambre []</i>) >
dans un deux étoiles	< (<i>gn_p (gn nomc (det indef sing)) (prep in)</i>), <i>obj.</i> (<i>hotel [(nbetoiles (entier 2))]</i>) >
à l'hôtel euh à l hôtel Caumartin	< (<i>gn_p (gn nomp (det def sing)) (prep a_prep)</i>), <i>obj.</i> , (<i>hotel [(identité (nom "Caumartin"))]</i>) >
pour y aller	< (<i>infinitif_p (prep pour)</i>), <i>obj.</i> , (<i>aller [(vers obj_contexte)]</i>) >
d ici	< (<i>gn_p (pronom lieu)(prep ab)</i>), <i>obj.</i> , <i>lieu_contexte</i> >
comment	< <i>adverbe, interrogation, comment</i> >
est-ce-que	< <i>expr, interrogation, oui_non</i> >
je peux	< (<i>gv 1 pres.</i>), <i>modalite, pouvoir</i> >
faire	< <i>infinitif, annexe, neutre</i> >

Tableau 3. Résultat de l'application des règles de niveau 1

5.2.5. Coordination des propositions et interprétation contextuelle

Cette dernière phase de l'analyse correspond au cas où l'énoncé se présente sous la forme de deux ou plusieurs propositions comme dans l'exemple développé dans les tableaux 3, 4 et 5. La coordination des propositions consiste à prendre en compte le

<p style="text-align: center;"><i>« je j ai réservé une chambre dans un deux étoiles »</i></p> <p>< (categs (gv 1 passe) (categs (gn nomc (det indef sing)) (gn_p (gn nomc (det indef sing)) (prep in))), obj., ((info présence) (de (réservation []) (de (chambre []) (hotel [(nbetoiles (entier 2))])))) ></p>
<p style="text-align: center;"><i>« à l'hôtel euh à l hôtel Caumartin »</i></p> <p>< (gn_p (gn nomc (det def sing)) (prep a_prep)), obj., (hotel [(identité (nom "Caumartin"))]) ></p>
<p style="text-align: center;"><i>« pour y aller d ici »</i></p> <p>< (infinitif_p (prep pour)), obj., (aller[(vers obj_contexte), (venant_de lieu_contexte)]) ></p>
<p style="text-align: center;"><i>« comment est-ce-que je peux faire »</i></p> <p>< (gv 1 pres), interrogation, comment ></p>

Tableau 4. Résultat de l'application des règles de niveau 2

<p style="text-align: center;"><i>« je j ai réservé une chambre dans un deux étoiles à l'hôtel euh à l hôtel "Caumartin" »</i></p> <p>< info, obj. > < présence, (de (réservation []) (de (chambre []) (hotel [(nbetoiles (entier 2)), (identité (nom "Caumartin"))])))) ></p>
<p style="text-align: center;"><i>« pour y aller d ici comment est-ce-que je peux faire »</i></p> <p>< interrogation, obj. > < comment, (aller [(vers obj_contexte), (venant_de lieu_contexte)]) ></p>

Tableau 5. Résultat de l'application des règles de niveau 3

contexte des propositions précédentes pour compléter la compréhension de la proposition concernée¹⁷. Deux traitements sont opérés :

– le premier consiste à résoudre des références. Ces 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 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 solution consiste à chercher dans le contexte quels objets peuvent sémantiquement correspondre à ces références. Ainsi, dans l'énoncé exemple, le lieu contextuel est la gare Saint-Lazare et le seul objet qui peut sémantiquement correspondre au *y* de « y aller » est l'hôtel Caumartin¹⁸. La représentation finale de l'énoncé est donnée dans le tableau 6 et la figure 6 ;

<pre> ((info présence) (de (réservation []) (de (chambre []) (hotel [(nbetoiles (entier 2)), (ident. (nom "Caumartin"))])))) ((interr. comment) (aller [(vers (hotel [(nbetoiles (entier 2)), (ident. (nom "Caumartin"))])), (venant_de (gare [(ident. (nom "St Lazare"))]))])) </pre>

Tableau 6. Représentation sémantique finale de l'énoncé exemple : formule logique

– le second traitement consiste à compléter les chaînes d'objets sans que cette complétion ne soit indiquée dans l'énoncé. L'analyse de l'énoncé « *j ai réservé une chambre à l hôtel Caumartin c'est quoi l'adresse* » rend deux propositions dans lesquelles les chaînes d'objets sont respectivement :

- 1 : (de (réservation [])
 (de (chambre []) (hotel, [(identité (nom "Caumartin"))])))
- 2 : (adresse [])

La deuxième chaîne d'objets sur laquelle porte la requête est complétée en
 (de (adresse []) (hotel, [(identité (nom "Caumartin"))])).

17. Cette mise en contexte utilise l'implication intuitionniste, l'un des connecteurs des formules manipulées par λProlog (formules de Harrop). Les objets des propositions précédentes sont dynamiquement mis en contexte pour la complétion de la proposition analysée.

18. Dans cette étude des dépendances entre propositions, 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.

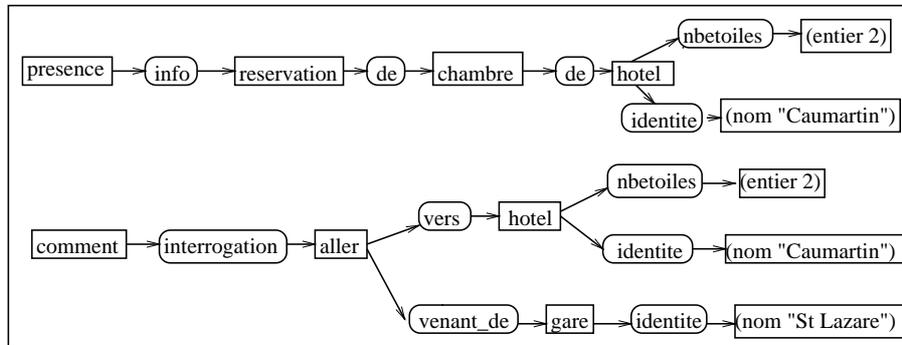


Figure 6. Représentation finale de l'énoncé exemple : graphe conceptuel

Le formalisme des chaînes d'objets permet d'énoncer un principe (dit *de complétion*) qui correspond à cette complétion : la chaîne d'objet sur laquelle porte la requête est complétée « à droite » par « un segment droit » (maximal d'un point de vue sémantique) de la chaîne d'objet contextuelle (cf. figure 7).

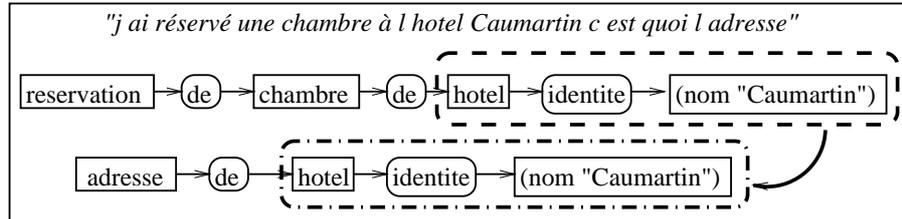


Figure 7. Complétion d'une chaîne d'objets

6. Quelques résultats

LOGUS-II a été soumis au test des 1 200 énoncés utilisés pour l'évaluation de LOGUS-I (cf. 2.5). Les résultats permettent de comparer les performances des deux systèmes.

Ces chiffres comparatifs sont indiqués dans le tableau 7 : les énoncés y sont classés dans la catégorie « compréhension incomplète » lorsque le sens général de l'énoncé a été dégagé mais qu'il y a eu omission d'un élément non essentiel (l'une des propriétés d'un objet par exemple). Les énoncés sont répartis en quatre séries de 300

énoncés chacune. Les séries 1 et 2 permettent essentiellement de mesurer la résistance du système face aux inattendus de l'expression orale ; les résultats obtenus par les deux systèmes y sont sensiblement égaux. La série 3 permet essentiellement de mesurer la couverture sémantique du système, au sens de la complexité des requêtes posées sur les objets concernés et des liens possibles entre ces objets (comparaisons entre objets, actes de langage variés : réfutation, acceptation, etc.). Entre les deux systèmes, ce type de couverture sémantique a été fortement augmentée et l'amélioration des résultats était donc attendue. La série 4 est composée d'énoncés complexes où les manifestations de l'expression orale sont poussées à l'extrême (très larges incises, reprises et expressions diverses). Ce sont essentiellement les résultats de cette série qui permettent de mesurer les réels progrès de l'analyse induits par la prise en compte d'éléments syntaxiques lors de la deuxième partie. Dégager le « sens » général d'énoncés aussi complexes est un résultat très encourageant.

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

Tableau 7. Résultats comparés de LOGUS-I et de LOGUS-II

Reco	Oral spontané : réparations	Oral spontané : ellipses, incises	Structures sémantiques inconnues (hors domaine)	Divers
10 %	20 %	26 %	35 %	9%

Tableau 8. Causes d'erreurs dans LOGUS-II

Un classement des différentes causes d'erreurs est donnée dans le tableau 8¹⁹.

Ces chiffres font apparaître que la portée des réparations (répétitions, reprises et auto-corrrections) reste l'une des causes essentielles des erreurs commises, avec les ellipses et les incises. La cause principale d'erreurs, les structures sémantiques inconnues, ne saurait être considérée comme redhibitoire, étant donné que les structures sémantiques acceptées par LOGUS-II sont déjà assez complexes et que ces énoncés seraient certainement considérés comme hors-domaine par un module de compréhension à structures sémantiques prédéfinies.

19. Les erreurs de reconnaissance qui rendent l'énoncé incompréhensible par élimination d'un concept essentiel ne sont pas prises en compte dans ces résultats (par exemple « *couteau* » pour « *coûte au* ») ; seul un couplage avec le module de reconnaissance permettrait de traiter ce type d'erreurs. Il va de soi qu'une telle interaction des deux modules pourrait être intéressante, mais elle n'est pas encore à l'ordre du jour.

7. Perspectives

Malgré les résultats encourageants obtenus par LOGUS, les travaux à réaliser demeurent nombreux et le plus important d'entre eux est d'étudier l'interdépendance du système avec les modules amont et aval : le module de reconnaissance et le module de dialogue.

Actuellement, les erreurs de reconnaissance sont traitées comme un bruit de fond auquel la robustesse de l'analyse doit permettre de faire face. Des études plus approfondies des effets de ces erreurs sont prévues ; elles permettront peut-être d'envisager des méthodes de détection et de traitements spécifiques.

Par ailleurs, la compréhension sous sa forme actuelle correspond à ce qu'il est convenu d'appeler *compréhension littérale*, même si l'étude des dépendances entre propositions constitue déjà une première prise en compte d'un contexte de dialogue. La frontière entre les tâches respectives du module de compréhension et du module de dialogue n'est pas clairement délimitée : pour certains, le module de dialogue doit orchestrer la compréhension et la compréhension littérale faite par le module de compréhension peut se réduire à une compréhension « aveugle ». Pourtant, certains énoncés ne peuvent se « comprendre » qu'en fonction du contexte et 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 l'avons définie : dans l'énoncé « *deux pour demain* » par exemple, aucun lien sémantique ne relie a priori les deux propriétés²⁰. Aussi, selon nous, la compréhension dite *contextuelle*²¹ ne se réduit pas à une simple adjonction à la compréhension littérale et elle doit donc devenir un élément essentiel de notre système. Si le formalisme adopté semble contenir les éléments nécessaires pour sa réalisation, il va de soi que sa mise au point demande certainement une étude plus précise des formes de dialogue que l'on souhaite pouvoir mettre en œuvre.

Prochainement, LOGUS-II doit participer à la campagne d'évaluation MEDIA dans le cadre du projet TECHNOLOGUE lancé par les ministères de la culture, de l'industrie et de la recherche. Cette évaluation doit utiliser le paradigme d'évaluation PEACE proposé par le LIMSI [DEV 02] et permettre de juger de la compréhension des énoncés oraux hors et en-contexte (de dialogue).

8. Conclusion

La compréhension automatique de la parole spontanée est un problème difficile, même dans les applications très circonscrites qui semblent actuellement les seules réellement envisageables. Pour obtenir des progrès significatifs, diverses disciplines

20. Ce problème est soulevé par Pierrel et Romary dans [PIE 00].

21. L'appellation est déjà en soi discutable puisque, si l'on en croit les linguistes, toute compréhension est contextuelle. La connaissance sémantique que nous utilisons fait également partie du contexte.

sont impliquées. Les linguistes sont évidemment concernés et ils nous décrivent des processus de communication et de compréhension complexes²².

Le système de compréhension LOGUS, que nous venons de décrire, présente une particularité essentielle : il utilise des formalismes logiques, grammaires catégorielles et graphes conceptuels, parfois considérés comme peu adaptés à la réalisation pragmatique de « véritables applications ». Selon nous, les deux points forts de ce formalisme sont les chaînes d'objets et la représentation en triplet des constituants. Les premières, très inspirées des graphes conceptuels, permettent une représentation sémantique à la fois simple et souple, sans cadres sémantiques prédéterminés et qui semble pouvoir aider à la compréhension contextuelle (le principe de *complétion* (cf. page 737) en est un exemple). Quant au double étiquetage syntaxique et sémantique, distinct de la traduction sémantique proprement dite, il permet de combiner les traitements syntaxique et sémantique tout en préservant une certaine généralité des règles. Les résultats prometteurs obtenus par LOGUS font penser que l'on peut concilier la recherche d'un certain formalisme avec l'efficacité.

Notons enfin, que la technique du *chunking* plutôt développée pour l'écrit est utilisée ici avec succès pour traiter la parole au prix de définition de *chunk* plus courts que pour l'écrit. Il convient en effet que les *chunks* soient contenus dans les structures locales qui respectent la norme de la langue. Ces structures sont évidemment plus petites à l'oral qu'à l'écrit, mais elles existent malgré tout à l'oral.

9. Bibliographie

- [ABN 91] ABNEY S., « Parsing by Chunks », BERWICK R., ABNEY S., TENNY C., Eds., *Principle Based Parsing*, Kluwer, 1991.
- [ALL 96] ALLEN J., MILLER B., RINGGER E., SIKORSKI T., « Robust Understanding in a Dialogue System », *34th annual Meeting of ACL*, 1996, p. 62–70.
- [ANT 99] ANTOINE J.-Y., GOULIAN J., « Le français parlé spontané est-il un langage à ordre variable ? », *Journées Internationales de Linguistique Appliquée, JILA*, 1999.
- [ANT 02] ANTOINE J.-Y., BOUSQUET-VERNHETTES C., GOULIAN J., KURDI M.-Z., ROSSET S., VIGOUROUX N., VILLANEAU J., « Predictive and Objective Evaluation of Speech Understanding : the "Challenge" Evaluation Campaign of the I3 Speech Workgroup of the French CNRS », *LREC, 3rd International Conference on Language Resources and Evaluation*, 2002.
- [AïT 02] AÏT-MOKHTAR S., CHANOD J.-P., ROUX C., « Robustness beyond Shallowness : Incremental Deep Parsing », *Natural Language Engineering*, vol. 8, 2002, p. 121–144.
- [AUS 62] AUSTIN J.-L., *How to do Things with Words*, Oxford, 1962.
- [BAN 99] BANGALORE S., JOSHI A. K., « Supertagging : An Approach to Almost Parsing », *Computational Linguistics*, vol. 25, n° 2, 1999, MIT Press.

22. Les études sur le dialogue faites par les linguistes concernent essentiellement des communications homme/homme. On peut penser que les énoncés émis par un humain envers une machine comportent moins d'implicite.

- [BAR 64] BAR-HILLEL Y., « *Language and Information* », chapitre On Categorical and Phrase Structure Grammars, Addison-Wesley, 1964.
- [BAR 01] BARGELLI D., LAMBEK J., « An algebraic Approach to French Sentence Structure », *Logical Aspects of Computational Linguistics*, Le Croisic, 2001, Springer, p. 62-78.
- [BEL 99] BELLEANNÉE C., BRISSET P., RIDOUX O., « A Pragmatic Reconstruction of λ Prolog », *Journal of Logic Programming*, vol. 41, 1999, p. 67–102.
- [BEN 02] VAN BENTHEM J., « *A Companion to Philosophical Logic* », chapitre Modal Logic, D. Jacquette, Blackwell, 2002.
- [BLA 90] BLANCHE-BENVENISTE C., *Le français parlé ; études grammaticales*, CNRS Editions, 1990.
- [BLA 02] BLANCHE-BENVENISTE C., « L'étude grammaticale des corpus de langue parlée en français », *Journées linguistiques de corpus*, 2002.
- [BUS 01] BUSZKOWSKI W., « Lambek Grammars Based on Prgroups », *Logical Aspects of Computational Linguistics*, Springer, 2001, p. 95–109.
- [DEL 01] DELSARTE P., THAYSE A., *Logique pour le traitement de la langue naturelle*, Hermes Science Publications, 2001.
- [DEV 02] DEVILLERS L., MAYNARD H., PAROUBEK P., « Méthodologies d'évaluation des systèmes de dialogue parlé : réflexions et expériences autour de la compréhension », *TAL*, vol. 43, n° 2, 2002, p. 155–184.
- [FRA 03] FRANK A., BECKER M., CRYSMANN B., KIEFER B., SCHÄFER U., « Integrated Shallow and Deep Parsing : TopD meets HPSG », *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, 2003, p. 104-111.
- [GAU 02] GAUVAIN J.-L. ET LAMEL L., « Systèmes de reconnaissance », MARIANI J., Ed., *Reconnaissance de la parole*, Hermès Sciences, 2002, p. 47–83.
- [GRI 89] GRICE H., *Studies in the Way of Words*, Harvard University Press, 1989.
- [HIR 98] HIRSCHMAN L., « Language Understanding Evaluations : Lessons Learned from MUC and ATIS », *1st Language Ressources and Evaluation Conference, LREC*, 1998, p. 117–122.
- [HYM 72] HYMES D., *Models of the Interaction of Language and Social Life*, Gumperz and Hymes, 1972.
- [KER 80] KERBRAT-ORECCHIONI C., *L'énonciation*, Colin, 1980.
- [KER 90] KERBRAT-ORECCHIONI C., *Les interactions verbales*, vol. 1, Colin, 1990.
- [LAM 58] LAMBEK J., « The Mathematics of Sentence Structure », *American mathematical monthly*, vol. 65, 1958, p. 154–169.
- [LAM 95] LAMEL L., BENNACEF S., BONNEAU-MAYNARD H., ROSSET S., GAUVAIN J.-L., « Recent Developments in Spoken Language Systems for Information Retrieval », *European Conference on Speech Technology (Eurospeech)*, 1995, p. 207–210.
- [LAM 00] LAMEL L., ROSSET S., GAUVAIN J.-L., BENNACEF S., GARNIER-RIZET M., PROUST B., « The LIMSI-ARISE System », *Speech Communication*, vol. 31, Elsevier, 2000, p. 339–353.
- [MIL 86] MILLER D., NADATHUR G., « Some Uses of Higher-Order Logic in Computational Linguistics », *24th Annual Meeting of the Association for Computational Linguistics*, 1986, p. 247–255.

- [MIN 96] MINKER W., BENNACEF S., « A Stochastic Case-Frame Approach for Natural Language Understanding », *International Conference on Spoken Language Processing (ICSLP)*, 1996.
- [MIN 99] MINKER W., *Compréhension automatique de la parole spontanée*, L'Harmattan, 1999.
- [MON 74] MONTAGUE R., « The Proper Treatment of Quantification in Ordinary English », THOMASON R., Ed., *Formal Philosophy*, Yale University Press, 1974.
- [PIE 00] PIERREL J.-M., ROMARY L., « Ingénierie des langues », chapitre Dialogue Homme-Machine, p. 331–349, Hermès, 2000.
- [RET 00] RETORÉ C., « Systèmes déductifs et traitement des langues : un panorama des grammaires catégorielles », *Technique et Science Informatique, numéro spécial de Traitement automatique du langage naturel (TAL)*, vol. 20, n° 3, 2000, p. 301–336.
- [SEA 70] SEARLE J. R., *Speech Acts*, Cambridge University Press, 1970.
- [SEN 92] SENEFF S., « TINA : a Natural Language System for Spoken Language Applications », *Computational Linguistics*, vol. 18, n° 1, 1992, p. 61–86.
- [SIM 03] SIMA'AN K., « *Recent Developments in Parsing Technology* », chapitre Robust Data Oriented Spoken Language Understanding, p. 323–338, Kluwer Publishers, Harry Bunt, John Carroll and Giorgio Satta édition, 2003.
- [SOW 84] SOWA J. F., *Conceptual Structures : Information Processing in Mind and Machine*, Addison-Wesley, 1984.
- [SOW 00] SOWA J., *Knowledge Representation*, Brooks/Cole Thomson Learning, 2000.
- [SPE 89] SPERBER D., WILSON D., *La pertinence : Communication et cognition*, Editions de Minuit, 1989.
- [SPE 00] SPERBER D., « La communication et le sens », MICHAUD Y., Ed., *Qu'est-ce que l'humain ?*, vol. 2, Odile Jacob, 2000, p. 119–128.
- [VAN 01] VANDERVEKEN D., « Universal Grammar and Speech Act Theory », VANDERVEKEN D., KUBO S., Eds., *Essays in Speech Act Theory*, John Benjamin, 2001, p. 25–62.
- [VIL 01] VILLANEAU J., ANTOINE J.-Y., RIDOUX O., « Combining Syntax and Pragmatic Knowledge for the Understanding of Spontaneous Spoken Sentences », *Logical Aspects of Computational Linguistics*, Springer, 2001, p. 279–295.
- [VIL 02] VILLANEAU J., ANTOINE J.-Y., RIDOUX O., « LOGUS : un système formel de compréhension du français oral spontané - présentation et évaluation », *TALN*, vol. 1, 2002, p. 165–174.
- [ZUE 00] ZUE V., SENEFF S., GLASS J., POLIFRONI J., PAO C. AND HAZEN T., HETHERINGTON L., « JUPITER : Telephone-Based Conversational Interface for Weather Information », *IEEE Transactions on Speech and Audio Processing*, vol. 8, n° 1, 2000, p. 85–96.