Revue québécoise de linguistique

Théorie syntaxique et théorie du parsage : quelques réflexions

Jean-Yves Morin

Volume 14, numéro 2, 1985

Linguistique et informatique

URI : https://id.erudit.org/iderudit/602537ar
DOI : https://doi.org/10.7202/602537ar

Aller au sommaire du numéro

Éditeur(s)

Université du Québec à Montréal

ISSN

0710-0167 (imprimé)
1705-4591 (numérique)

Découvrir la revue

Citer cet article

Morin, J.-Y. (1985). Théorie syntaxique et théorie du parsage : quelques réflexions. *Revue québécoise de linguistique*, *14*(2), 9–48. https://doi.org/10.7202/602537ar

THÉORIE SYNTAXIQUE ET THÉORIE DU PARSAGE : QUELQUES RÉFLEXIONS*

Jean-Yves Morin

1. Prologue

Depuis la parution du livre de Marcus (1980), il est redevenu de bon ton, en linguistique informatique, quand il est question de parsage, de faire référence à une théorie syntaxique «établie» (e.g. théorie des traces (Marcus 1980), grammaire lexicale fonctionnelle (Church 1982), théorie gouvernement-liage (Wehrli 1983), grammaire syntagmatique généralisée (Thompson 1981, 1982, 1983)).

Ainsi, Marcus, dans sa thèse, terminée en 1977 mais publiée en 1980, fait référence à plusieurs concepts de la *théorie standard étendue*, entre autres, celui de *trace*. De plus, il tente de fournir une explication opératoire

---

* J'ai choisi d'employer sans ambages le terme «parsage» pour «analyse syntaxique (sémantique) automatique». On ne peut guère m'y reprocher un anglicisme puisqu'il s'agit d'un latinisme en anglais même («parse» ‹ «pars orationis»). Je me suis donc permis cet emprunt au latin. D'ailleurs, pourquoi ne pas voir dans ce terme bien pratique, à suffixe très québécois, une résurgence de l'ancien français *parsuir* «développer, défaire» (qui donnerait quelque chose comme «parsuissage», il est vrai).

Je tiens à remercier les participants au séminaire de «Linguistique informatique» et au cours de «Grammaires formelles et applications» de l'Université de Montréal au printemps 1984, Eric Wehrli, de UCLA, qui, le premier, m'a fait entrevoir l'intérêt du problème du parsage pour la théorie syntaxique, ainsi que Jacques Labelle, rédacteur de la *Revue québécoise de linguistique*, dont la patience a été mise à dure épreuve par mes atermoiements. Inutile d'insister sur le fait que je suis seul responsable de toutes les erreurs qui ont pu se glisser ici ainsi que du ton, parfois un peu trop didactique, de ces réflexions. Je demande l'indulgence du lecteur là-dessus.

On présupposera une connaissance, au moins sommaire, de la théorie de la grammaire syntagmatique généralisée (GSG). Cf. Gazdar (1982), Gazdar-Pullum (1982) et Morin (1984) pour une présentation d'ensemble de la théorie, de même que Evans & Gazdar (1984) pour la description détaillée d'un environnement informatique pour GSG basé sur Prolog, dans le contexte du système multilangages POPLOG.

Cette recherche a été subventionnée par le CRSH dans le cadre du projet «Modèle formel d'acquisition de la syntaxe» (subvention CRSH-410-82-0801).

de deux principes de la théorie standard étendue : la *condition de sujet spécifié* et le principe de *subjacence*, en les faisant découler du fonctionnement même de son analyseur, qui ne dispose que de structures de données limitées : une pile de nœuds à compléter («mères» à la recherche de «filles»), dont chacun est associé à un (ou plusieurs) ensemble(s) de règles et un tampon (ou file d'attente) de constituants à rattacher à ces nœuds («orphelines» en quête d'une «mère»). Cf. figure 1. On comprend sans peine l'intérêt que les hypothèses de Marcus ont pu soulever tant en linguistique théorique qu'en intelligence artificielle.

Du point de vue de la théorie linguistique, il s'agit de la première tentative sérieuse de tenir compte des principes les plus abstraits de la théorie standard étendue dans un parseur d'une certaine envergure. En ce sens, le travail de Marcus constitue, pour la linguistique informatique, ce que celui de Lightfoot (1979) constitue pour la syntaxe diachronique, ou ceux de Baker (1979, 1981 réd.) pour le problème logique de l'acquisition : une extension des principes les plus abstraits de la théorie à des domaines nouveaux et extrêmement riches. Ce n'est d'ailleurs pas un accident que toutes ces «ouvertures» soient strictement contemporaines l'une de l'autre.

Quant au point de vue de l'intelligence artificielle, la contribution de Marcus y est, à notre avis, encore plus révolutionnaire. Allant à l'encontre des théories et pratiques établies, il propose que des *mécanismes computationnellement restreints* (mais linguistiquement riches) sont suffisants pour faire l'analyse syntaxique des langues naturelles.

Alors que la plupart des modèles informatiques de la syntaxe des langues naturelles (réseaux de transition augmentés ou «ATN», systèmes-Q, grammaires de métamorphoses, etc.) emploient des ressources de calcul théoriquement illimitées, le modèle de Marcus est extrêmement parcimonieux dans son utilisation de telles ressources. Comme on l'a déjà mentionné, il n'utilise qu'une pile de nœuds incomplets (associés à des paquets de règles actives) et un tampon de mots ou constituants en attente d'être rattachés (pile de «mères» et tampon de «filles» ou d'«orphelines», pour reprendre la métaphore ci-haut). Malgré les dissemblances évidentes, le modèle de Marcus emprunte beaucoup à l'algorithme d'Earley (algorithme général d'analyse des langages indépendants du contexte, cf. infra) ainsi qu'à divers autres modèles d'analyse des langages indépendants du contexte. Ceci n'a rien d'étonnant, dans la mesure où la théorie du parsage

des langages de type 2 (indépendants du contexte) est particulièrement développée, ce qui ne constitue pas un accident d'ailleurs.

La figure 1 présente schématiquement les structures de données et les composantes majeures du modèle de Marcus.

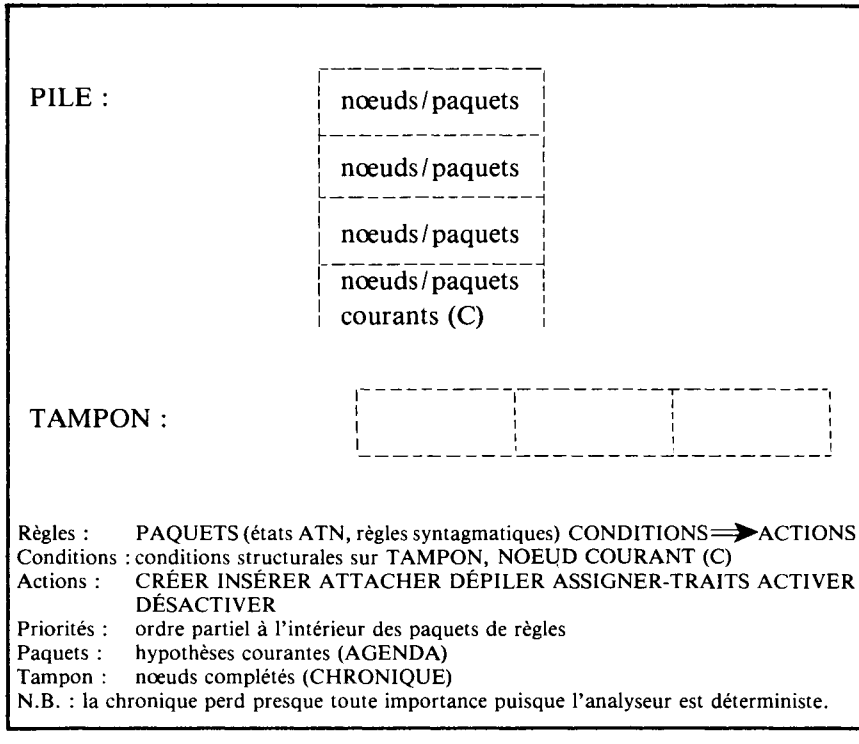


Figure 1. PARSIFAL, Structure et propriétés générales du modèle.

Dans le Parsifal de Marcus (1980), la condition de sujet spécifié découle d'un principe opératoire que l'auteur nomme *contrainte gauche-à-droite* («L-to-R constraint»). Cette «contrainte» stipule que :

> **Contrainte gauche-à-droite**
> Les constituants en attente (dans le tampon, cf. figure 1) doivent être *presque toujours* rattachés à des nœuds supérieurs dans l'ordre dans lequel ils apparaissent dans le tampon.

En d'autres termes, un constituant C_I doit être rattaché à un nœud supérieur avant un constituant C_J si $I < J$ (les indices correspondent à la

«case» du tampon qu'un constituant occupe avant d'être rattaché à un nœud supérieur). La clausule d'exception («presque toujours») prévoit les cas où le constituant rattaché le premier n'occupe pas la position 1 mais la position 2 du tampon, e.g. les cas d'inversion de l'auxiliaire en anglais où le SN sujet occupe la position 2 mais doit être rattaché au P supérieur, situé dans la pile, avant l'auxiliaire occupant la position 1 (cf. figure 2).

Pile de nœuds actifs

...

P/ ... (paquets de règles actifs)

Tampon

1 : AUX : (Is)
2 : SN : (John)
3 :

Suite non encore vue par l'analyseur
(coming?)

Figure 2. Configuration instantanée lors de l'analyse d'une phrase à inversion en anglais.

Cependant, si l'on y regarde d'un peu plus près, on se rend compte que la contrainte gauche-à-droite ne produit l'effet de condition de sujet spécifié que par accident (i.e. les sujets se trouvent les SN les plus à gauche, *en général*, en anglais) et qu'en fait, elle est entièrement ad hoc.

Par ailleurs, à l'intérieur de la théorie standard étendue, la condition de sujet spécifié (prise, en quelque sorte, comme un absolu par Marcus) a été remplacée, d'abord par la condition d'opacité, puis, à l'intérieur de la théorie gouvernement-liage, par le jeu complexe de conditions de liage définies à partir de notions comme celles de gouvernement, de domaine de gouvernement, elles-mêmes définies à partir de celles de tête, de c-commande et de projection. Il est évident que la contrainte gauche-à-droite, étroitement liée à certains faits concrets de l'anglais, ne peut certainement plus constituer une «explication» de principes aussi abstraits. Une situation analogue se présente dans le cas de l'«explication» du principe de subjacence. Ce principe est plus ou moins stipulé tel quel dans le fonctionnement de l'analyseur. Il n'est donc guère étonnant qu'il en «découle».

Ainsi donc, on se rend compte, à l'examen approfondi, que ces références à la théorie syntaxique prennent plus la forme d'analogies superficielles entre les représentations produites ou de stipulations plus ou moins ad hoc des principes de la théorie dans l'analyseur que d'intégration organique (et d'interaction enrichissante de part et d'autre) entre théorie syntaxique et théorie (et pratique) du parsage[1].

Néanmoins, Marcus a ouvert une boîte de Pandore en essayant de réétablir des liens étroits entre théorie syntaxique et théorie du parsage. De tels liens, plusieurs autres chercheurs ont également essayé d'en tracer récemment : Wehrli (1983, 1984), Berwick (1982, 1983) et Berwick-Weinberg (1982, 1983, 1984) pour la théorie gouvernement-liage (GL); Church (1982) et Ford-Bresnan-Kaplan (1983) pour la grammaire lexicale fonctionnelle (GLF) et Thompson (1981, 1982, 1983), Shieber (1983, 1984), Gazdar (1984), Bear (1982), Bear-Karttunnen (1979) et plusieurs autres pour la grammaire syntagmatique généralisée (GSG).

C'est la trame de ces liens, aussi intéressants pour la théorie de la syntaxe que pour celle du parsage, que nous essaierons de faire ressortir dans ce travail, écrit du point de vue de la théorie de la *grammaire syntagmatique généralisée*.

2. Le problème du parsage

Le problème du parsage consiste à trouver une procédure qui permette de traduire une représentation r_I (une chaîne de mots, élément d'un

1. Les travaux basés sur la théorie des *grammaires syntagmatiques généralisées* échappent plus facilement à ce type de critique. Cette théorie, en tant qu'extension conceptuelle de celle des grammaires context-free, possède déjà un modèle formel et un modèle informatique. La théorie du parsage des langages context-free est extrêmement riche (cf. Sheil 1976). C'est donc uniquement dans ce cadre théorique que l'on peut parler, autrement que métaphoriquement, d'implications réciproques et d'interactions rigoureuses entre théorie syntaxique et théorique du parsage.

Cependant, la plupart des travaux inspirés de GSG en linguistique informatique sont encore inédits ou difficilement accessibles. De plus, ils se contentent souvent d'emprunts locaux (métarègles, schémas DI/PL, etc.) sans exploiter toute la richesse du formalisme et de la théorie elle-même (on songe en particulier à la théorie généralisée des catégories et à son modèle formel, cf. cependant Evans et Gazdar 1984). Enfin, à notre connaissance, GSG n'a jamais encore été confrontée (sauf évidemment dans nos propres travaux non publiés) aux problèmes de description du français. Aussi semble-t-elle encore peu connue dans le monde francophone.

C'est le but de ces quelques notes que d'attirer l'attention sur certains points de la théorie du parsage des langues naturelles qui reçoivent un éclairage nouveau du fait de l'adoption d'une version très riche (bien que restrictive) de la théorie syntagmatique généralisée.

ensemble C) en une ou plusieurs autres représentations r_J (r_K, r_L, etc.), élément(s) d'un (ou plusieurs) ensemble(s) R plus riche(s) et mieux «manipulable(s)» que la première, i.e. sur laquelle (lesquelles) les traitements (e.g. vérification d'erreurs, sémantique, déduction, traduction, etc.) pourront s'effectuer directement, et de la façon la plus cohérente possible, ce qui n'est pas le cas avec une simple chaîne de mots.

Parsage : C $\Longrightarrow$ R
r_I $\Longrightarrow$ r_J (r_K, r_L, etc.)

Les *arborescences* (étiquetées) sont particulièrement intéressantes à cet égard puisqu'elles constituent une structure de données riche et souple du point de vue informatique, familière et expressive pour le linguiste. De plus, les propriétés des systèmes formels qui les engendrent, les grammaires syntagmatiques ou indépendantes du contexte («context free grammars») sont bien connues et computationnellement aussi bien que linguistiquement intéressantes (cf. Sheil 1976 et Pullum 1983). Enfin, comble de bonheur, ce type de représentation correspond directement à une propriété syntaxique fondamentale des langues naturelles : la *structure syntagmatique.*

Dans des théories comme la théorie gouvernement-liage (GL) ou la grammaire lexicale fonctionnelle (GLF) la structure syntagmatique (S-structure dans GL et C-structure dans GLF) joue un rôle central mais moins important que dans la théorie de la grammaire syntagmatique généralisée (GSG), dans la mesure où l'on retrouve dans ces théories (GL et GLF) plusieurs autres types (ou niveaux) de représentation (D-structure et «forme logique» pour la théorie GL, F-structure pour la théorie GLF).

En grammaire syntagmatique généralisée, la structure syntagmatique (ou S-structure) constitue le *seul niveau de représentation syntaxique.* On ne saurait trop insister sur ce point, essentiel du point de vue de la théorie du parsage[2]. Comme GSG est une théorie monostratale (i.e. à un seul niveau de représentation syntaxique), le parsage y constitue une application quasi-fonctionnelle (à cause de l'existence de chaînes ambiguës). Le problème du parsage peut donc y être vu simplement comme celui de la définition d'une (quasi-)fonction de transformation (transduction) d'une chaîne c_I (élément d'un ensemble L) en un arbre a_I, (élément de l'ensemble A(L)) étiqueté par

2. Cf. Gazdar (1982) pour des arguments en faveur de cette représentation «monostratale».

des catégories complexes (généralisées) représentant sa structure syntagmatique (enrichie) :

$$\text{Parsage} : L \Longrightarrow A(L)$$
$$c_I \Longrightarrow a_I$$

3. Les problèmes du parsage

Ayant rappelé en quoi consistait *le* problème du parsage en GSG, on peut se tourner vers *les* problèmes auxquels la définition d'une telle procédure doit faire face en ce qui concerne les langues naturelles.

3.1 *Complexité espace/temps*

Pour qu'une procédure d'analyse syntaxique ait des chances de survie il faut qu'elle n'utilise que des ressources limitées autant en ce qui concerne l'espace qu'elle occupe en mémoire pendant le traitement (i.e. son espace de «résidence» mais surtout l'espace de travail qu'elle exige pour être opérationnelle) qu'en ce qui concerne le temps (i.e. la suite d'opérations, relativement complexes) qu'elle nécessite. Il y a souvent corrélation entre les deux, ou encore échange. Deux problèmes sont à considérer : la complexité des grammaires et des lexiques utilisés et celle des structures de travail nécessaires pour que l'analyseur fonctionne efficacement.

Les premiers analyseurs syntaxiques pour les langues naturelles (e.g. l'analyseur prédictif de Kuno (1966) ou celui de la MITRE) contenaient des milliers de règles complexes et étaient très lents (de l'ordre de 20 secondes pour les phrases les plus simples ou de quelques minutes pour une phrase d'une dizaine de mots). Certains progrès ont été accomplis dans ce domaine, on ne peut cependant guère les comparer aux progrès dans la sophistication (et la baisse foudroyante des prix) du matériel informatique. Ainsi, l'analyseur expérimental EQSP (Martin et al. 1981), qui possède une assez bonne «couverture» (cf. infra pour une définition plus précise de ce terme) de l'anglais, comporte un peu plus d'une centaine d'états; il s'agit d'un analyseur utilisant les réseaux de transition augmentés ou ATN. Le programme-source compte 87 pages de code LISP sans compter le lexique. Sa performance est assez bonne : de l'ordre de quelques secondes pour une phrase relativement complexe.

On possède actuellement peu de données expérimentales ayant une réelle valeur pratique en ce qui concerne l'efficacité des différents schèmes de compression des grammaires pour le parsage des langues naturelles, dans

la mesure où la plupart des systèmes implantés ont des lexiques très restreints (de l'ordre de quelques centaines à un millier de mots), alors que l'on peut présumer, sans grand risque d'erreur, que les lexiques constitueront les modules les plus lourds, tant par leur volume que par les temps d'accès qu'ils entraîneront dans un analyseur sophistiqué. Pour le moment, il ne peut s'agir que de spéculations, mais il est intéressant de noter que le formalisme «modulé» de la grammaire syntagmatique généralisée permet des compressions étonnantes dans le volume des grammaires, tout en maintenant la précision (et le type formel). Shieber (1984) a démontré que le formalisme DI/PL se prêtait à une utilisation directe dans un analyseur, sans qu'on ait à générer l'ensemble des règles de la grammaire-objet. On peut conjecturer qu'il en est de même des métarègles (cf. cependant Thompson (1982) pour une opinion contraire) et des principes d'instanciation. Plusieurs chercheurs travaillent actuellement sur ces problèmes (entre autres à SRI, autour de Shieber) et les premiers résultats sont extrêmement encourageants.

En ce qui concerne les structures de travail nécessaires au fonctionnement harmonieux de l'analyseur, elles dépendent en grande partie de la façon dont celui-ci parcourt l'espace des hypothèses de structuration de la chaîne. Un analyseur en parallèle (cf. infra, § 3.1) peut être très rapide, du moins en théorie, mais il occupera généralement beaucoup d'espace inutilement puisqu'il mène de front toutes les hypothèses et doit donc toutes les stocker. Un analyseur en série (cf. infra, ibidem) occupera moins d'espace (toujours en théorie, toutes choses étant égales par ailleurs, ce qui n'est d'ailleurs jamais le cas en pratique) mais sera plus lent en moyenne s'il doit considérer plusieurs «chemins» et faire marche arrière («backtrack») fréquemment. Il est cependant possible de contourner le dilemme en adoptant des structures de données riches qui allient les avantages des deux approches et en utilisant au maximum les propriétés «diagnostiques» des items lexicaux des langues naturelles (cf. Marcus 1975 et Small 1980). On tentera donc d'éviter l'«explosion» par une utilisation judicieuse de *représentations* riches et souples.

Au niveau de la théorie mathématique de la *complexité* du calcul, il existe, pour les grammaires syntagmatiques, un bon nombre de preuves de limitation (i.e. bornes supérieures en fonction de la longueur de la chaîne à analyser). Cependant, la plupart de ces résultats n'ont que peu d'intérêt pratique, dans la mesure où ils ne portent que sur des cas limites jamais

atteints par les grammaires réelles. Par exemple, comme le fait remarquer Perrault (1983), aucun langage CF «réel» connu n'exige des ressources plus que linéaires, aussi la limite supérieure $O(n^3)$ pour les algorithmes généraux de reconnaissance des langages CF (CKY, Cocke-Kasami-Younger, ou Earley) n'a-t-elle qu'un intérêt purement théorique[3].

En fait, on pourrait croire que les langues naturelles, avec leurs imposants facteurs d'ambiguïté et d'indéterminisme, seraient peut-être les premiers exemples de langages CF pour lesquels ces limites théoriques soient effectivement pertinentes. Church et Patil (1982) (cf. aussi Martin, Church et Patil 1981) s'intéressent à ce problème d'un point de vue à la fois théorique et expérimental. Ils ont découvert, en expérimentant avec l'analyseur EQSP, que certains sous-langages de l'anglais étaient systématiquement ambigus (cf. infra) et produisaient un comportement (un nombre d'analyse) en fonction quasi-exponentielle («catalane») de la longueur de la chaîne à analyser. Cependant, il faut noter que ce qui est vrai d'un sous-langage ne l'est pas nécessairement du langage lui-même. En particulier, la constante «grammaire-système» *O* est beaucoup plus importante qu'ils ne le laissent voir. On reviendra, à la section suivante, sur certaines de leurs hypothèses, trouvailles ou conclusions concernant l'ambiguïté de certaines constructions et la croissance quasi-exponentielle dans l'espace ou le temps qu'elles produisent.

Du point de vue qui est le nôtre, celui de la réalisation d'une théorie de la grammaire en un analyseur «réaliste», ces problèmes sont tangentiels et ne doivent pas nous préoccuper à priori. Ce qui ne veut pas dire que certaines hypothèses que nous proposerons n'auront aucun effet sur le

3. Il n'en reste pas moins que seules les grammaires CF permettent de tels résultats et garantissent une reconnaissance en temps linéaire dans la plupart des cas. Aussi la GSG, en tant que système de type CF est la seule théorie pour laquelle de tels résultats existent, nonobstant les affirmations de Berwick (1982) et Berwick-Weinberg (1982), qui s'acharnent à répéter qu'il n'est pas impossible que l'on arrive à de tels résultats ou à des résultats encore plus significatifs si l'on formalise adéquatement la théorie gouvernement-liage. C'est là un truisme autant pour la théorie GL que pour la syntaxe fonctionnelle de Martinet ou la psychosystématique guillaumienne. Il n'est, de toute évidence, pas *impossible* que l'on arrive à de tels résultats. En attendant une formalisation de GL (ou de la syntaxe fonctionnelle, ou de la psychosystématique), seule GSG permet d'en parler sérieusement, puisque seule elle possède un modèle proprement formel.

Noter aussi la limite «absolue» $O(n^2, {}^{81})$ obtenue par Valiant (1975) à l'aide des techniques de multiplication de matrices. Cf. Martin et alii (1981) pour une discussion de ces résultats dans un contexte d'application concrète (parseur pour l'anglais), le système EQSP.

«rendement» de l'analyseur mais bien que ces questions sont pour nous, linguistes, relativement secondaires.

3.2 *Ambiguïté (combinatoire)*

Dès les premières recherches en vue de la traduction automatique (cf. Kaplan 1950, Kuno-Oettinger 1963), on a remarqué que l'un des problèmes les plus difficiles de l'analyse syntaxique automatique des langues naturelles découle de l'ambiguïté de celles-ci. En effet, les mots, les expressions, les syntagmes, les propositions exprimées dans une langue naturelle ont une fâcheuse tendance à être interprétables de plusieurs façons différentes. Malgré cela, on remarque que les êtres humains perçoivent très rarement ces ambiguïtés dans un contexte donné et que même, hors de tout contexte apparent, elles ne semblent leur causer aucune difficulté d'interprétation alors qu'elles font, pour ainsi dire, exploser l'espace/temps de travail de l'analyseur.

Tout aussi étonnant que cela puisse paraître, il y a très peu d'études portant spécifiquement sur ce problème. Outre les contributions de Kaplan (1950) et de Kuno-Oettinger (1963) mentionnées ci-haut, citons, en linguistique générale, le livre de Kooij (1971), les sections 6.1.3. et 6.6.2. de Lyons (1968) et 10.4 de Lyons (1977), les paragraphes 427 et 561-563 (entre autres) de Bally (1944) et enfin, d'un point de vue computationnel, la thèse de Boguraev (1979). Aussi me permettrai-je d'établir certaines distinctions, qui me semblent fondamentales, mais qui ne font pas partie, que je sache, de l'univers conceptuel commun.

Au niveau le plus élémentaire, on dira qu'une chaîne est ambiguë si elle possède plusieurs représentations grammaticales (pour un même niveau ou une même strate, dans les théories polystratales). Prenons l'exemple classique :

(1) La belle ferme le voile.

Au niveau de la S-structure (seul niveau de représentation, rappelons-le, dans une théorie monostratale comme GSG), on peut attribuer à cette chaîne les structures suivantes :

(1) a. [P [SN [DET la] [ADJ belle] [N ferme]] [SV le voile]]
b. [P [SN [DET la] [N belle]] [SV [V ferme] [SN le voile]]]

c. [P [SN [DET la] [N belle] [*ADJ* ferme]]
[SV le voile]]

La structure (1a) correspond à l'interprétation la plus probable hors contexte, la structure (1b) à une interprétation un peu plus rare, enfin, la structure (1c) à une interprétation à peu près «inacessible» sans contexte ni explication pour un locuteur du français.

Distinguons tout d'abord entre ambiguïté lexicale et ambiguïté structurelle (la seconde pouvant découler de la première comme dans notre exemple). Il y a ambiguïté lexicale lorsqu'une même forme (minimale) appartient à plusieurs catégories lexicales différentes. Ainsi, dans notre exemple, «belle» peut être adjectif ou nom, «ferme» peut être nom, verbe ou adjectif, «le» peut être déterminant ou pronom et «voile» peut être verbe ou nom. Il y a ambiguïté structurelle lorsqu'une même chaîne peut être représentée structurellement de plus d'une façon. C'est le cas de la chaîne (1) ci-haut et de la chaîne (2) ci-dessous (où l'ambiguïté structurelle ne découle pas d'une ambiguïté lexicale) :

(2) Deng a collé l'étiquette sur la bouteille sur la table dans la cuisine.

En fait, l'ambiguïté lexicale n'est qu'un cas particulier (et particulièrement intéressant dans la mesure où il est prédictible) de l'ambiguïté structurelle.

On doit également distinguer entre ambiguïté locale et ambiguïté globale. Une ambiguïté locale existe dans une portion de la chaîne mais n'est pas nécessairement compatible avec l'ensemble des analyses bien formées de la chaîne, i.e. ne contribue pas nécessairement à l'ambiguïté globale de la chaîne. Par exemple, dans (1), le mot «la» (comme le mot «le») est localement ambigu puisqu'il peut être aussi bien déterminant que pronom. Cependant, la catégorie *pronom* est incompatible avec l'une quelconque des analyses du mot suivant («belle»). L'ambiguïté est donc rapidement éliminée. En fait, on se rend compte que des 48 (= 2 x 2 x 3 x 2 x 2) représentations devant découler de l'ambiguïté de chacun des termes (ambiguïtés locales), seules 3 sont globalement bien formées.

Cette distinction est particulièrement importante dans la mesure où ce sont souvent les ambiguïtés locales qui font croître exponentiellement l'espace de recherche de l'analyseur. De plus, ces ambiguïtés risquent de

passer totalement inaperçues dans un modèle de compétence, d'être «transparentes», alors que leur élimination rapide est essentielle pour une «performance» adéquate de l'analyseur. D'ailleurs, même au niveau des ambiguïtés globales, les phrases nettement ambiguës pour un locuteur-auditeur humain sont extrêmement rares et, dans ce cas, on leur attribue tout de suite un seul sens ou un ensemble restreint de sens. Il semble que ce soit l'ambiguïté elle-même qui soit difficile à percevoir pour l'être humain. Voici par exemple une phrase hautement ambiguë pour un analyseur standard (e.g. un ATN non déterministe) :

(3) La petite ferme la porte sur le feu du four de la cuisinière dans la cuisine du château de madame.

Cette phrase ne pose que très peu de problèmes d'interprétation (si même elle en pose) à un francophone. Il est même difficile d'apercevoir les dizaines d'ambiguïtés qu'on devrait y retrouver. Comme autre exemple, on pourrait citer les phrases suivantes du corpus de Malhotra (1975) :

(4) What was the change in total manufacturing cost from 1972 to 1973? (69)

(5) I mean I would like the cost of each product broken down on a direct and indirect basis. (130)

(6) The intent of my question is to find out if you know if your accounting methods can relate the changes in sales to changes in your expense structures. (510)

(7) In as much as allocating costs is a tough job I would like to have the total costs related to each product. (958)

Les nombres entre parenthèses indiquent le nombre d'analyses différentes que l'analyseur EQSP, mentionné ci-haut, a fourni pour chacune de ces phrases. Ainsi, la phrase (7), qui ne semble absolument pas ambiguë reçoit 958 analyses syntaxiques différentes (ce qui ne prend que 6,5 secondes, étant donné certaines astuces de programmation et la sophistication du matériel sur lequel tourne EQSP). La question de l'élimination de ces centaines d'analyses parasites (résidus d'ambiguïtés locales, qui ne devraient pas passer au niveau global) se pose avec une acuïté toute particulière.

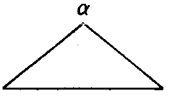
Par ailleurs, même en ce qui concerne les ambiguïtés globales réelles (comme en (1) ci-dessus), il semble que le locuteur-auditeur humain puisse se créer «sur le tas» des scénarios-contextes désambiguïsants, qui guident l'analyse et, fils d'Ariane, l'empêchent de «dédaler» en vain. On pourra

essayer de tenir compte de ces «contraintes sémantiques (pragmatiques) adjuvantes» dans la conception d'un analyseur. Cependant, nous ne nous intéresserons ici qu'aux aspects strictement syntaxiques de l'analyse.[4]

On peut chercher à identifer, dans la structure syntaxique des langues naturelles, les zones spéciales, productrices d'ambiguïtés et donc génératrices d'explosions combinatoires. Cette analyse fonctionnelle des constructions génératrices d'ambiguïtés et de leurs propriétés spécifiques permettra d'établir des stratégies (les plus générales possibles), de façon à résoudre directement, à contourner ces problèmes ou à s'y résigner (si l'on peut démontrer qu'ils sont insolubles).

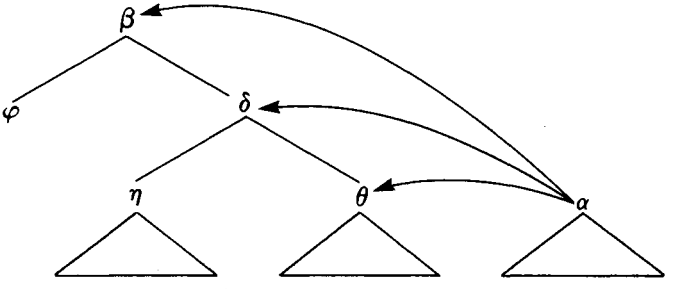
Logiquement, l'ambiguïté peut porter sur trois aspects de la structure syntagmatique :

a) L'*étiquetage* (par une catégorie complexe) d'un constituant déja trouvé.

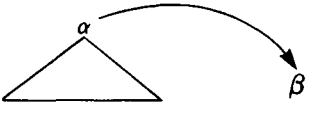


CAT (α) = ?

b) Le *rattachement* d'un constituant à un autre.



c) La *fermeture* d'un constituant.



On peut déjà noter que la fermeture est l'inverse du rattachement. Ainsi, un constituant α est d'autant plus générateur d'ambiguïtés d'analyse qu'il peut être étiqueté différemment (a), qu'il a de points de rattachement possibles

4. Cf. Boguraev (1979) pour un exemple détaillé de système utilisant un tel filtrage sémantique. Voir aussi Ritchie (1981).

(b) ou qu'il peut accepter de filles (c). En termes de grammaire syntagmatique (généralisée ou non), ceci signifie que l'on peut calculer le cœfficient d'ambiguïté d'une grammaire directement à partir de celle-ci. On peut donc, à partir de ce repérage, créer des «diagnostics», qui permettent de pratiquer une certaine «prophylaxie» dans l'analyse (i.e. d'éviter les problèmes).

Pour une langue comme le français, les sources d'ambiguïté de type (a) sont essentiellement dues à des ambiguïtés lexicales (beaucoup moins nombreuses en français qu'en anglais)[5]. On connaît l'exemple célèbre «Le pilote ferme la porte» (cf. aussi l'exemple (1) ci-haut), où chacun des termes est ambigu et où pourtant la phrase n'a que deux interprétations possibles (dont l'une beaucoup plus probable que l'autre) au lieu des 48 (2 x 2 x 3 x 2 x 2) localement possibles mais pour la plupart globalement mal formées. La pratique montre que de tels exemples sont assez rares et qu'un analyseur (bien) guidé par des hypothèses permet d'éviter de parcourir inutilement l'ensemble des hypothèses possibles pour chaque mot.

Dans un parseur basé sur la théorie GSG, les ambiguïtés d'étiquetage locales ne posent pas de problème puisqu'elles portent sur des valeurs non spécifiées de traits dans l'arbre d'analyse. On n'a donc pas à reproduire toutes les valeurs possibles mais à les introduire au moment où elles deviennent disponbiles. Il n'y a pas d'explosion exponentielle de l'espace de recherche et les ambiguïtés locales demeurent «transparentes» (ou «invisibles»), comme il se doit. Les seules ambiguïtés de ce type qui pourraient poser des problèmes un peu sérieux seraient des cas où un item lexical pourrait être soit tête, soit spécificateur (e.g. «avoir», «être», «faire» qui peuvent être auxiliaires (spécificateurs) ou verbes principaux (têtes)) et

5. Cf. Morin (1984) pour une présentation plus complète de cette hypothèse, que Wehrli (1984) adopte sans la mentionner, et pour cause... Wehrli (1983, 1984), le monsieur Jourdain de GSG, dans son analyseur pour le français censément basé sur la théorie GL décrit un analyseur purement syntagmatique, sans transformations mais avec l'équivalent de métarègles, qui reconstruit une S-structure tout à fait compatible avec GSG, mais assez loin de GL, malgré son titre («A Government-Binding Parse for French»). À partir de la tête, un réseau de transition non récursif de droite à gauche (cf. infra) rattache les spécificateurs de cette tête (qui sont en attente dans un tampon). Les spécificateurs ne constituant jamais des catégories récursives, un réseau de transition simple (mais néanmoins augmenté de conditions) permet donc de les analyser. Quant aux compléments (catégories récursives), ils sont rattachés conformément aux prédictions générées par la tête. Il est vrai que le parseur de Wehrli n'est pas monostratal puisqu'il reconstruit aussi une structure fonctionnelle en tous points conformes à celles de GLF (et tout à fait différente de la «forme logique» de GL).

où cette ambiguïté (les têtes sont immédiatement projetées sur une catégorie syntagmatique idoine, pas les spécificateurs, cf. infra §3) ne serait pas éliminable localement. À notre connaissance, de tels cas n'existent pas en français; quand ils semblent exister, comme dans les exemples de Marcus (1980) :

(8) Have the students who failed the exam taken the make-up?
(9) Have the students who failed the exam take the make-up?

le problème n'est qu'apparent puisque les auxiliaires de l'anglais *doivent* être projetés comme des V (cf. Gazdar, Pullum & Sag 1982 et Bear 1982).

Les ambiguïtés de type (b) et (c) sont beaucoup plus fréquentes et posent des problèmes extrêmement difficiles. Elles sont dues à l'omnivalence de certaines constructions, en particulier les SP (sélectionnés ou non par une tête), les coordonnées et, dans une moindre mesure, les propositions subordonnées (qui ne posent que des problèmes assez triviaux et dont nous ne parlerons donc pas ici). On sait que les SP peuvent apparaître, en nombre indéfini, dans l'expansion de chacune des catégories (syntagmatiques ou propositionnelles) du français selon le schéma suivant (cf. Morin 1984) :

(d) $X' \rightarrow \ldots X \ldots SP^*$

Généralisation que l'on peut d'ailleurs exprimer au moyen d'un métaschéma :

(e) $\alpha \rightarrow ß \Longrightarrow \alpha \rightarrow ß, SP^+$

Quant aux *structures coordonnées*, elles peuvent apparaître à tous les niveaux (y compris le niveau lexical), selon le schéma suivant (cf. Gazdar et al. (1982) pour plus de détails et la décomposition de ce schéma en deux (un schéma binaire et une fermeture positive : CAT' → CAT', CAT' et CAT' → CAT', CAT'$^+$) selon la valeur des x_I) :

(f) CAT' → CAT'[CONJ x_0], CAT'[CONJ x_1]*

C'est le principe de réalisation des conjoints (PRC) qui assure un certain degré de parallélisme entre les conjoints en stipulant qu'ils doivent tous être des extensions de leur «mère» (cf. Gazdar-Pullum (1982) ou Morin (1984) pour la définition formelle de ce concept). De plus certains de leurs membres peuvent constituer des fragments (que nous noterons *p*-). C'est le cas particulièrement dans les constructions «scindées» («gapping») :

(10) Xiao-bo buvait du maotai et Yi-ning __ du Coca-Cola.

(11) [P [P [SN Xiao-bo] [SV [V buvait] [SN du maotai]]] [P- [CONJ et] [P- [SN Yi-ning] [SN du Coca-Cola]]]]

Comme une discussion des problèmes de parsage associés aux structures coordonnées nous entraînerait trop loin, nous nous contenterons d'illustrer les problèmes d'ambiguïté à partir du rattachement des SP. Lorsque l'analyseur a ouvert un SP, le problème du rattachement de celui-ci se présente. On a les cas de figure suivants :

— Le SP est sélectionné par la tête d'un constituant, qui n'est pas encore saturé et qui est minimal, i.e. θ dans le schéma (b) ci-haut. C'est le cas le plus simple. Le rattachement à θ et la saturation de celui-ci (pour le SP) se fait directement. Comme le nombre de SP pouvant être sélectionnés est restreint, on atteint vite la saturation complète. Il y a donc très peu de problèmes dans ce cas.

— Le SP est sélectionné par la tête d'un constituant non encore saturé mais non minimal, e.g. ß ou ó ci-dessus. Ce cas est un peu plus complexe, car α pourrait être rattachable à θ et ce serait un autre SP, disons α qui viendrait satisfaire la sélection de ß ou ó.

— Enfin, le SP peut n'être sélectionné par aucune tête. Il n'y a donc pas de critère local pour un rattachement ou l'autre.

Plusieurs chercheurs se sont penchés sur ce problème depuis l'article de Kimball (1973). Nous ne pouvons que mentionner ici les travaux de Berwick et Weinberg (1982), Cowper (1976), Crain et Fodor (1984), Fodor (1978, 1983, 1984), Fodor et Frazier (1980), Frazier (1979), Frazier et Fodor (1978), Ford et al. (1983) et Wanner (1980) dont l'approche est plutôt psycholinguistique et ceux de Boguraev (1979), Church (1982), Church et Patil (1982), Marcus et al. (1983), Martin et al. (1981), Pereira (1982), Sager (1973, 1980) et Wehrli (1984), dont l'approche est plus strictement computationnelle et donc plus pertinente à notre propos actuel.

Il n'est pas dans intention de faire l'examen des différentes hypothèses qui ont été proposées pour rendre compte de l'attachement des SP. On peut cependant distinguer trois grandes approches :

— Attachement fixe (et minimaliste) : *forme normale la plus à droite.* Dans le cadre des grammaires d'extraposition, Pereira (1982) propose que les modificateurs nominaux (il ne considère que ce cas particulier) soient rattachés en forme normale au SN dont la tête est la plus à droite (donc

minimale et proximale). Son traitement formel est très intéressant mais manque de généralité au niveau linguistique (i.e. il est trop lié, dans ses contraintes, à l'ordre des mots de l'anglais et ne traite, comme nous l'avons dit, que des SP dépendant d'une tête nominale). On en retiendra surtout l'idée qu'une forme normale est peut-être suffisante à l'interprétation sémantique. En fait, ce qui est nécessaire au fonctionnement de l'interpréteur sémantique, c'est un arbre qui contienne toute l'information nécessaire à l'interprétation, pas nécessairement l'arbre le plus proche de cette interprétation (qui, de toute façon, est structurée assez différemment). On peut considérer ce type de rattachement comme heuristique, en ce sens qu'il utilise une stratégie générale sans avoir à en calculer a priori toutes les conséquences. Cf. aussi l'attachement minimal dans Church (1982) et les heuristiques «chronologiques» de Wanner (1980) de même que les travaux à orientation psycholinguistique cités plus haut.

— Multi-attachement : rattachement du SP à tous les constituants (non saturés). C'est l'hypothèse la plus ancienne (cf. Sager (1973), qui parle d'«ambiguïté permanente prédictible») et la plus répandue (pseudo-attachement de Church (1982), «ambiguïté universelle» de Martin et al. (1981) et de Church-Patil (1982), «oracle sémantique» de Marcus (1980)). Comme le notent Church-Patil (1982) cette stratégie de rattachement équivaut en un certain sens à du liage différé : on rattache à tout en attendant de pouvoir faire un choix (sémantique) spécifique. Les mêmes auteurs notent que dans le cas d'une ambiguïté systématique et «permanente» (i.e. non éliminable syntaxiquement), il peut être plus profitable de générer directement l'ensemble des rattachements possibles que de l'énumérer. En effet, on peut remarquer que le nombre de points de rattachement d'une séquence de *n* SP (après une tête nominale) est égal au n IÈME nombre catalan. Les nombres catalans (d'après le nom du mathématicien E. Catalan, rien à voir avec les ambiguïtés dans les séquences de clitiques du catalan) constituent une série à la croissance quasi-exponentielle dont les premiers termes sont 1, 1, 2, 5, 14, 42, 132, 469, 1430, 4 862, etc. On peut donc prévoir (et appliquer) directement les Cat_N rattachements possibles en une seule étape.

— D-attachement : c'est l'approche adoptée par Marcus et al. (1983). Il s'agit non pas de fournir un rattachement spécifique mais une description des contraintes sur les rattachements possibles (la D-théorie est la théorie des descriptions d'arbre). Elle se situe dans la ligne de l'hypothèse de

l'*indélibilité* des choix d'analyse de Marcus (1980). Il s'agit de remplacer le prédicat de domination immédiate servant à définir le rattachement par un prédicat de domination. L'ensemble des prédicats de domination ayant un nœud donné (e.g. α dans notre schéma ci-haut) comme (deuxième) argument constitue l'espace de rattachement de ce nœud. Par exemple, on pourrait avoir l'ensemble de contraintes suivant (toujours en référence aux étiquettes du schéma (b) ci-dessus) :

$$\{\text{DOMINE } (\beta, \alpha), \text{DOMINE } (\delta, \alpha)\}$$

α ne peut donc être rattaché qu'à δ ou à θ (puisqu'un rattachement direct à β violerait la deuxième contrainte : DOMINE (δ, α)). Ces contraintes peuvent être fournies par l'analyseur à différents moments de l'analyse (e.g. aux différents moments ou α devient lié par les propriétés de sélection des têtes qui le précèdent) ou découler d'options par défaut (ce que Marcus et al. nomment des descriptions standard). Il pourrait d'ailleurs être très profitable d'expérimenter avec diverses descriptions standard. Malgré sa complexité (plus apparente que réelle) et certaines obscurités (aussi bien dans le texte de Marcus et al. que dans notre présentation trop succinte), cette approche semble très intéressante, du moins dans ses grandes lignes. En effet, elle apparaît un peu comme une synthèse des deux approches précédentes.

Comme la stratégie précédente, cette stratégie équivaut à du liage différé. Dans le multi-attachement, on rattache de façon universelle avant de faire un choix spécifique, dans le d-attachement, on contraint l'*espace* des rattachements possibles en les «décrivant» tant qu'on ne peut pas préciser un (ou plusieurs) rattachement(s) «classique(s)». D'autre part, comme la première stratégie, elle permet des options par défaut et des choix peu coûteux syntaxiquement et facilement récupérables sémantiquement.

Ajoutons, pour terminer ce survol trop hâtif, que GSG fournit un formalisme très riche pour définir des contraintes de rattachement. Outre le gouvernement (au sens GSG, cf. Morin (1984) : dépendances locales des ajouts envers leur tête), on a les dépendances locales non gouvernementales (qui n'ont pas la tête pour origine, cf. Gazdar-Pullum (1982) pour le «control agreement principle»[6]) et surtout les dépendances globales

6. Cf. Gazdar-Pullum (1982) sur la «Head Feature Convention», le «Foot Feature Principle» et le «Control Agreement Principle». Il est remarquable de noter que Colmerauer (communication personnelle), à partir d'un point de vue tout différent (les grammaires de

exprimées au moyen de traits (traits de PIED, CORNE, ACCORD, QU, etc., cf. Bear (1982), Gazdar et al. (1982), Gazdar-Pullum (1982) et Morin (1984)).

D'autre part, comme les étiquettes des nœuds de l'arbre que cherche à construire l'analyseur sont des catégories complexes (≠ symboles monadiques) et qu'elles n'ont pas à être entièrement spécifiées, e.g. CAT[1] est une étiquette parfaitement bien formée, représentant n'importe quelle catégorie de niveau 1 (i.e. syntagmatique, en fait, il s'agit d'une abréviation pour [CAT [NIVEAU 1]]), on peut y exprimer à la fois les sélections (= prédictions) qui montent de la tête (et de certains spécificateurs) que les saturations apportées par le rattachement des différents ajouts. On peut donc assurer un flot d'échanges harmonieux entre ce que Kaplan (1975) nomme «producteurs» («producers») : prédictions des nœuds à compléter ou «mères» en quête d'une progéniture et ce qu'il nomme «consommateurs» («consumers») : nœuds à rattacher ou «filles» en quête d'une «mère».

3.3 *Imprécision et non déterminisme*

3.3.1 Imprécision

On a également remarqué que les grammaires des langues naturelles avaient tendance à contenir des règles floues, imprécises, indéterministes. Par exemple, les catégories syntaxiques semblent constituer des sous-ensembles flous. Un infinitif n'est-il pas tout aussi nominal que verbal, un participe tout aussi adjectival que verbal? Qu'est-ce qu'une préposition, un verbe transitif, un Comp, un quantificateur, un pronom clitique?

Il en est de même, du moins en apparence, pour les règles. Comment peut-on réécrire SV, SN, SP? Quelle est la structure des SP non sélectionnés? Quelle est la différence entre une relative sans tête et une interrogative indirecte, entre une relative et une clivée? Comment interprète-t-on les relatives? Et, à un niveau plus abstrait, qu'est-ce qu'un réfléchi, un liage, une catégorie vide?

Ces questions, qu'on ne se pose guère lorsque l'on fonctionne à l'intérieur d'un paradigme fermé, mais qui sont pourtant fondamentales, deviennent inévitables dans la mesure où tous ces choix doivent être

---

clauses définies et de métamorphoses enchâssées en Prolog), arrive à des mécanismes très proches de ces principes. Cf. aussi note 10 infra.

Cf. Bear-Karttunen (1979) pour les représentations employées dans la chronique.

spécifiés précisément dans un analyseur et, si possible, justifiés par des considérations empiriques plutôt que purement pratiques.

Encore une fois, GSG permet de formuler des hypothèses très précises (et aisément testables expérimentalement) à ce sujet. Avec son riche système de catégories (généralisées), qui peuvent être plus ou moins spécifiées, la théorie GSG permet d'établir des distinctions très fines entre (sous-)catégories et d'utiliser ces distinctions pour guider directement l'analyse. Par exemple, le mécanisme de l'indexation, sous sa forme généralisée, où les items lexicaux I ne sont pas indexés relativement aux éléments de l'ensemble des règles R_M mais aux éléments de sous-ensembles désignés (par la tête de règle) X_I de l'ensemble $P(R_M)$ des parties de R_M permet une expression précise et efficace des propriétés de gouvernement de I, qui guide de façon étroite les prédictions générées au cours de l'analyse.

3.3.2 Non déterminisme

Il me semble aussi que l'analyse d'une phrase d'une langue naturelle ne puisse se faire de façon localement déterministe, i.e. qu'en un point de l'analyse, il soit difficile, sinon impossible de choisir quel «chemin» emprunter sans aller voir beaucoup plus loin ou sans «assumer ses erreurs» et revenir en arrière.

C'est à Marcus (1980) que revient le mérite d'avoir attiré l'attention des linguistes sur ce problème et d'avoir montré qu'il n'était peut-être pas impossible d'analyser de façon déterministe les langues naturelles, en exploitant au maximum les contraintres locales qu'elles contiennent. Cependant, l'idéalisation de Marcus, qui élimine arbitrairement toute ambiguïté lexicale est fort discutable, de même que ses emprunts sporadiques à la théorie des traces. On peut également critiquer la confusion que son modèle entretient entre règles de grammaire et stratégies d'analyse, confusion que les travaux de Berwick (1981, 1982) accentuent encore.

3.4 *Économie*

Un autre problème dans l'analyse syntaxique des langues naturelles, qui découle des problèmes précédents, est celui de l'économie. Comme la procédure d'analyse risque d'être longue et complexe, de se heurter à des ambiguïtés et de devoir parcourir plusieurs chemins différents à partir d'un même point ou faire marche arrière, il s'avère important d'économiser ses efforts, en d'autres termes :

— ne faire que ce qui est utile (nécessaire)
— ne pas refaire une analyse déjà faite.

Pour résoudre ce problème de gaspillage d'énergie-temps, on devra se munir de structures de contrôle et d'archivage complexes du type agenda (contrôle) et/ou chronique (archivage).

3.5 *Couverture*

Les langues naturelles sont beaucoup plus riches que les langages artificiels tant dans la variété de leur vocabulaire que dans les constructions possibles. Les mots, les catégories, les structures et les interprétations possibles sont plus nombreux (de plusieurs ordres de grandeur) dans les langues naturelles que dans les langages de programmation.

Aussi, si l'on veut un analyseur qui soit autre chose qu'un jouet-prototype, on devra s'intéresser au problème de la couverture, i.e. de la dimension relative du sous-ensemble d'une langue qu'il peut couvrir. Ce problème est loin d'être trivial et la plupart des analyseurs qui tournent actuellement sont très limités de ce point de vue. On remarque qu'aussitôt qu'un prototype est étendu ou généralisé à un domaine plus large à un niveau quelconque, ses «adhocités» tendent à croître de façon exponentielle et forment des métastases dans chacun des modules du système. Le problème est d'autant plus sérieux qu'en tant que linguistes, nous cherchons à définir des propriétés générales des langues naturelles et non pas des trucs d'analyse ad hoc qui fonctionnent à tout prix.

On s'est longtemps contenté, en linguistique informatique, d'une approche que je qualifierais d'«empiriste» au problème de la couverture. On définit un système très général, avec lequel on expérimente, soit sur des textes (c'est l'approche du système DÉREDEC, cf. Plante (1981)), soit sur des phrases-tests définies par le concepteur ou existant dans la littérature linguistique (cf. Marcus (1980) ou Wehrli (1984) pour un excellent exemple de cette approche). Parfois, on peut tenter d'identifier des domaines où les problèmes d'analyse (syntaxique et/ou sémantique) soient plus restreints. On retrouve cette approche aussi bien dans le système SHRDLU de Winograd (1972), que dans LUNAR (Woods et al. 1972) et dans les différents systèmes de traduction automatique élaborés par le groupe TAUM (TAUM-Météo et TAUM-Aviation). Elle a donné naissance à ce que l'on peut désigner comme le paradigme des sous-langages (cf. Kittredge-Lehrberger 1981). Si une telle approche peut être très efficace

dans certains cas bien particuliers, elle ne doit pas nous cacher le problème de la couverture qu'elle écarte plutôt qu'elle ne le résoud.

En fait, on pourrait s'étonner que le problème de la couverture n'ait été que rarement attaqué de front. Pourtant, il n'y a rien là d'étonnant. L'approche empiriste est typiquement celle du praticien qui doit concevoir un système qui «tourne» et n'a pas à se poser de questions de linguistique générale. Quant aux linguistes, dans le cadre du paradigme de la grammaire générative (au sens large), ils ne se préoccupent pas de ce problème. Comme dans n'importe quelle discipline scientifique, ce n'est pas tant la description exhaustive d'un objet donné (système solaire, anatomie du chimpanzé, etc.) que l'explication de certaines de ses propriétés significatives qui constitue l'objet de la recherche. Ce n'est pas la description de toutes les phrases d'une langue (ou d'un sous-langage) qui constitue le but des travaux d'un linguiste génératiViste mais bien l'explication de certaines propriétés très abstraites du langage (ou de la grammaire, cf. Chomsky 1980).

Cependant, comme le disait Saussure, «la langue est un système où tout se tient» et il y a un intérêt non seulement pratique mais également théorique à étendre la portée descriptive de la théorie ne serait-ce que pour éviter le piège béant de la circularité spéculative.

Pour ce faire, on doit combiner analyse expérimentale et analyse fonctionnelle. Malhotra (1975) s'est livré à une analyse expérimentale des besoins dans le domaine des bases de données de gestion. Il a demandé à des gestionnaires d'essayer son système «parfait» d'interrogation de bases de données de gestion en langue naturelle (anglais). En fait, c'est Malhotra lui-même, avec ses assistants de recherche, qui interagissait avec eux. L'expérience a permis de recueillir un important corpus d'interrogations «naturelles» de même que des commentaires sur le comportement du «système» par un échantillon représentatif de la clientèle-cible. La principale conclusion que l'on peut tirer de ce travail est qu'un système doit disposer de toute les ressources syntaxiques d'une langue naturelle pour être «convivial», autrement, mieux vaut utiliser un langage formel de requête ou un système à menus sophistiqués. C'est aussi la conclusion de Tennant et al. (1983), qui ont construit un tel système. L'approche «sous-langage» semble donc exclue, du moins au niveau syntaxique. Il serait en tout cas intéressant de se livrer à des expériences analogues pour le français. Les mythes véhiculés par les médias sur les «systèmes intelligents» ne pourront que nous faciliter la tâche (pour une fois).

Mais, en tant que linguistes, ce qui nous apparaît comme plus intéressant, c'est une analyse fonctionnelle, c'est-à-dire un inventaire des «objets linguistiques» et de leurs propriétés essentielles (cf. aussi Winograd (1983), Appendice B).

3.5.1 Objets linguistiques (constructions syntaxiques)

N'importe quel système un peu sophistiqué devra pouvoir reconnaître les constructions suivantes :

- Impératives
- Interrogatives (binaires et focales)
- Comparatives
- Coordonnées
- Relatives (avec et sans tête, avec et sans «relateur»)
- Complétives (tensées et infinitives)
- «Circonstancielles» (finales, temporelles, consécutives, conditionnelles, etc.)
- Éléments périphériques :
  - Clivées :
    C'est *dans «Peter-le-Noir»* que les bouteilles intactes constituent un
  - Disloquées :
    La cocaïnomanie de Sherlock Holmes, le docteur Watson la tolérait sans l'approuver.
  - Topicalisées :
    Au non-aboiement du chien, Holmes comprend que le coupable n'est autre que la victime __.
  - Extrapositions :
    Holmes donne un ordre à Watson.
    ?Holmes donne à Watson un ordre.
    ??Holmes donne l'ordre de se taire et d'examiner les gestes de Moriarty à Watson.
    Holmes donne à Watson l'ordre de se taire et d'examiner les gestes de Moriarty.
  - Appositions :
    Jeudi le 28 novembre, le jour de la paye, un incendie criminel s'est déclaré à la caisse Papineau.
- Expressions idiomatiques
- Clitiques

Cet inventaire de constructions essentielles ressemble à s'y méprendre à un inventaire des questions en suspens ou du moins des problèmes difficiles en grammaire générative. Au moins peut-on dire que le traitement des interrogatives (Engdahl 1982, Gazdar 1982, Gazdar-Pullum-Sag 1982), des comparatives (Gazdar 1980, Klein 1981), des relatives (Engdahl 1982), des complétives (Klein et Sag 1982) et des coordonnées (Gazdar et al. 1982) en GSG est assez complet, de même que celui de la topicalisation (Gazdar 1982, Engdahl 1982). Les expressions idiomatiques commencent à être étudiées sérieusement (Wasow et al. 1982). Restent les impératives, les «circonstancielles», les clitiques et toutes les autres constructions à «éléments périphériques». Les impératives ne devraient pas poser trop de problèmes. Il s'agit d'un ensemble très limité de formes ne pouvant apparaître que dans des non enchâssées. Quant aux «circonstancielles», elles nécessitent une étude approfondie des propriétés lexicales et syntagmatiques des prépositions, qui est loin d'être faite. Pour les clitiques, cf. Gazdar (1982), Morin (1984) et Morin (à paraître) «Les clitiques dans la théorie généralisée des catégories». Université de Genève, juin 1984. Les constructions à «éléments périphériques» ont été un peu négligées en grammaire générative. Il semble néanmoins que le formalisme DI/PL et la théorie des traits permette une approche éclairante de ce type de phénomène (cf. aussi Huot-Lemonnier 1984).

3.5.2 Propriétés syntaxiques

Tout analyseur syntaxique doit être à même de produire des représentations rendant compte des propriétés suivantes.

— S-structure

La structure syntagmatique doit évidemment être représentée dans l'output de l'analyseur.

— M-structure

Jusqu'à un certain point, la structure interne des mots doit également être représentée dans l'output (ou du moins être codée au lexique)

— D-structure

L'ensemble des dépendances (locales et globales, gouvernementales et non gouvernementales) doit pouvoir être représenté.

— gouvernement
  — sélection d'arguments
  — rection de la forme des arguments présents
— dépendances locales non gouvernementales
  — accords (e.g. spéc ⟹ tête)
  — filtrage, haplologie (e.g. *[de de], * [SX [qu]] que], etc.)
— dépendances globales
  — anaphore
  — contrôle
  — accord
  — quantification
  — liage-filtrage («traces»)

etc.

En GSG, le formalisme DI/PL, associé aux métarègles et métaschémas, aux principes d'instanciation et à la théorie généralisée des catégories permet de représenter aussi bien la S-structure que la D-structure (enchâssée dans les étiquettes de la S-structure). La M-structure relève de la théorie du lexique sur laquelle de plus en plus de travaux (aussi bien en GSG qu'en GL ou en GLF) se concentrent. On devrait donc bientôt disposer d'un formalisme intégré où les problèmes de couverture se poseraient plutôt en termes sémantiques et lexicaux qu'en termes proprement syntaxiques.

4. Stratégies

Pour faire face à ces problèmes, on peut employer diverses stratégies (certaines constituent plutôt des tactiques, mais nous négligerons cette distinction, plus de niveau que de principe, pour le moment). Certaines de ces stratégies ont été créées pour faire face aux problèmes de compilation des langages de programmation, d'autres sont plus spécifiquement orientées vers l'analyse des langues naturelles. Nous n'en ferons ici qu'un bref inventaire indicatif et non exhaustif.

4.1 *Parallèle/série*

Une analyse en parallèle considère toutes les hypothèses à la fois et les mène de front. Une telle approche a l'avantage d'éviter de refaire des analyses déjà faites. Elle fournit l'ensemble des solutions possibles. Cependant, comme on l'a déjà noté, le fait de devoir conserver toutes les hypothèses tout au long de l'analyse «bouffe» littéralement l'espace de

travail de la machine. De plus, on peut ajouter qu'il ne semble pas que les humains fonctionnent systématiquement ainsi (et ce sont d'excellents parseurs).

Une analyse en série prend chaque hypothèse une à une et la (les) pousse jusqu'à leurs limites. En cas de cul-de-sac, on fait marche arrière jusqu'au dernier point où un choix était possible et on essaie un autre membre de l'alternative. Si on ne cherche qu'une analyse, quelle qu'elle soit, de la chaîne, cette méthode peut être très efficace. Elle ne gaspille pas indûment d'espace avec des analyses inutiles. Si les hypothèses sont bien ordonnées (e.g. les plus probables d'abord) le comportement moyen d'un analyseur en série sera nettement supérieur à celui d'un analyseur en parallèle (toutes choses étant égales par ailleurs). Cependant, si la langue à analyser possède un haut facteur d'indéterminisme (i.e. quantité de choix équiprobables), et il semble bien que ce soit souvent le cas des langues naturelles, une telle approche «bouffera» en temps ce que l'approche parallèle bouffe en espace et plus encore. De plus, les nombreux retours en arrière exigeront d'être gérés par une structure de contrôle extrêmement complexe.

Il est donc très difficile d'opter, en l'absence d'autres arguments pour une stratégie ou pour l'autre. Il peut être préférable de choisir une combinaison des deux types de stratégies. On pourrait proposer que l'analyse procède en parallèle pendant quelques instants (i.e. quelques nœuds, quelques mots) et qu'elle passe ensuite au mode sériel (et déterministe). Il semble d'ailleurs que les être humains fonctionnent un peu ainsi.

Notons que l'alternative parallèle/sérielle correspond à une exploration en largeur («breadth first») ou en profondeur («depth first») de l'espace des analyses possibles.

4.2 *Descendante/ascendante*

Dans la recherche d'une analyse, on peut procéder soit en faisant des hypothèses que l'on teste peu à peu en les confrontant aux données, soit en examinant les données elles-mêmes et en élaborant les hypothèses d'analyse à partir d'elles. Dans le premier cas, on adopte une stratégie descendante ou guidée par des hypothèses, dans le second, une analyse ascendante ou guidée par les données. Si les données permettent de générer des prédictions (e.g. les propriétés de gouvernement des items lexicaux), on peut parler

d'une stratégie hybride de coin gauche (à la fois guidée par les données et par des hypothèses que celles-ci permettent de générer). Cette dernière stratégie semble particulièrement bien adaptée au traitement des langues naturelles (cf. Bear 1982, Bear-Karttunen 1979 et Slocum 1984 pour des exemples d'analyseurs à stratégie «angulaire»).

Si l'on conçoit l'analyse comme la construction d'une arborescence pour une chaîne de mots donnée, on voit facilement à quoi les termes d'«ascendant», de «descendant» et de «coin gauche» peuvent faire référence. Une analyse (syntagmatique) est descendante si elle tente de construire l'arbre à partir de la racine (i.e. des hypothèses les plus abstraites, e.g. P' → Comp P ou P → SN SV). Elle est ascendante si elle tente de le construire à partir de ses feuilles (i.e. des mots concrets apparaissant dans la chaîne). Elle est de coin gauche si elle le construit à partir de sa feuille la plus à gauche de façon ascendante et descendante.

Une analyse descendante ne propose que des combinaisons «utiles» (i.e. qui peuvent aboutir à des P complètes), alors qu'une analyse ascendante combine aveuglément les constituants qu'elle retrouve. Cependant, une analyse descendante va engendrer des hypothèses ne correspondant à rien dans la chaîne, ce qu'une analyse ascendante ne fera jamais.

Descendante	**Ascendante**
Reste + position + grammaire	Combinaison + position + grammaire
⟹ règles utilisables	⟹ règles utilisables
partie droite → reste	partie gauche → combinaison

Les deux types d'approche ont donc leurs avantages et leurs inconvénients. Le mérite d'Early (1968) est d'avoir montré qu'elles n'étaient pas incompatibles du moins pour les langages CF. Rappelons que la théorie GSG étant un enrichissement de celle des grammaires CF, tous les résultats et algorithmes concernant celles-ci y sont applicables et/ou utilisables.

On cherchera donc à combiner l'astuce des analyses descendantes avec la sécurité des analyses ascendantes. Il s'agira d'éviter les culs-de-sac de part et d'autre en combinant adroitement les deux types d'analyses. La richesse lexicale des langues naturelles, largement exploitée dans les systèmes du type «mots-experts» (Small 1980), guidera les aspects ascendants de l'analyse, alors que les contraintes locales et les «espérances» engendrées tant par les «mots-experts» que par les règles en guideront les aspects descendants. La notion de diagnostic (cf. Marcus 1980) peut être utile à ce niveau, de même que celles de TÊTE, de SPÉCificateur et de COMPLément.

En fait, la stratégie optimale semble être de se servir de la tête comme pivot et générateur d'hypothèses. En GSG, toute tête a une projection immédiate. Si l'on ajoute à cela la contrainte que toute projection syntagmatique a une tête lexicale (S-structure «plate»), avec laquelle elle partage toutes ses propriétés, sauf celle de niveau évidemment, on peut construire un algorithme étonnamment performant. En effet, la saisie d'une tête X entraîne immédiatement la construction du X' correspondant (si X est ambiguë, X' le sera aussi, momentanément puisqu'elle porte tous les traits de X).

[j [j-1 [X [alpha]]]] ⟹ [j [i [X' [alpha]]]]

où *j* représente la position du mot de catégorie X, le premier indice d'un élément de la chronique (la structure d'archivage des structures déjà trouvées au cours de l'analyse, cf. infra) est son extrémité droite (la position où il se termine, *j* dans le cas présent), le second est celle de la fin du constituant précédent (*j-1* pour un item lexical) et la double flèche est interprétable comme une relation «si-alors».

On peut donc immédiatement ajouter à la chronique (la liste des constituants trouvés avec leur situation dans la chaîne) un X' dont l'extrémité gauche (indice *i*) sera celui du spécificateur de X (propriété lexicale) libre le plus à gauche dans la chronique. Tous les spécificateurs de X libres dans la chronique (et compatibles avec les traits de X, évidemment) peuvent donc être immédiatement rattachés à X' (ils ne seront donc plus libres).

Par ailleurs les prédictions de X' (qui correspondent aux propriétés de gouvernement de X) sont empilées, par ordre de préséance, dans l'agenda (la liste des constituants recherchés) et l'analyse peut se poursuivre avec le mot suivant.

4.3 *Déterministe/non déterministe*

Une stratégie d'analyse syntaxique est déterministe au sens strict si, pour tout point dans la chaîne d'entrée, le choix d'une analyse est strictement guidé par l'état de l'analyseur en ce point et par les informations qu'il peut y obtenir localement. C'est sur la définition de localité que joue Marcus avec ce qu'il appelle son hypothèse de déterminisme. En fait, il s'agit plutôt d'une hypothèse d'indélibilité, i.e. tout choix posé à un certain point de l'analyse est irréversible, ce qui n'empêche pas d'utiliser de

l'information non locale pour opérer lesdits choix. Il ne s'agit donc pas de déterminisme au sens strict.

Cependant, il est intéressant de voir que les langues naturelles ne sont peut-être pas si indéterministes qu'on pourrait le penser de prime abord. Dans la plupart des cas «normaux», Marcus (1980) suggère que l'analyse syntaxique peut procéder de façon quasi-déterministe, avec une «fenêtre» limitée à au plus trois éléments. Si les analyses de Marcus s'avèrent défectueuses de plusieurs points de vue, son approche générale n'en demeure pas moins très intéressante et pourra nous servir dans la construction d'un analyseur restrictif pour le français.

4.4 *Ponctuelle/périscopique*

Une stratégie d'analyse peut également être ponctuelle ou périscopique dans la mesure où elle a (ou non) accès à de l'information située dans une portion de la chaîne non encore analysée. Un analyseur linéaire de «gauche à droite» peut ainsi disposer d'un périscope lui permettant d'examiner certains aspects de la chaîne situés à la droite de sa position présente. En général, le périscope est limité à un nombre maximum de «mots» (e.g. analyseurs lr(K)) ou de constituants (e.g.tampon du Parsifal de Marcus à trois fenêtres de constituants).

4.5 *Directionalité*

Un autre aspect de la stratégie à adopter dans la construction d'un analyseur concerne la direction à adopter dans l'exploration de la chaîne d'entrée. La stratégie non marquée consiste à la parcourir de gauche à droite, ce qui correspond à l'ordre d'écriture dans les langues indo-européennes modernes. On pourrait également la parcourir de la fin au début (de droite à gauche) ou même en boustrophédon (alternativement de gauche à droite puis de droite à gauche) selon les problèmes qu'elle pose (i.e. cheminement unidirectionnel) pour les chaînes faciles à analyser, boustrophédon pour les autres).

On pourrait également rechercher dans la chaîne certains îlots de certitude à partir desquels on pourrait construire la structure simultanément vers la gauche (e.g. spécificateurs) et vers la droite (e.g. compléments). Cette stratégie est très employée dans les systèmes de reconnaissance de la parole.[7]

7. Cf. Bates (1978), Walker (1978 réd.) et Carroll (1982) sur les ATN à «îlots».

On a vu que le type de description proposé ici (GSG restrictive) se prête assez bien à ce genre d'analyse. Il faut noter que l'analyseur avec lequel nous expérimentons actuellement circule strictement de gauche à droite dans la chaîne mais de droite à gauche dans la chronique et que les prédictions sont empilées dans un agenda indépendant. Ce qui nous amène à toucher un mot des mécanismes de contrôle.

4.6 *Contrôle (et archivage)*

Les mécanismes de contrôle employés pour guider l'analyse constituent un autre paramètre important à prendre en considération. On peut vouloir tenir une chronique des étapes déjà franchies, des constituants déjà trouvés, etc., de façon à éviter les répétitions et le travail inutile. Suite à Bear (1982) et Bear-Karttunen (1979), nous adoptons pour la chronique une structure de liste où chacun des éléments est de la forme suivante :

[j [i [χ]]]

j est un indice correspondant à l'extrémité du constituant
i est un indice correspondant à sa frontière gauche
x est une variable sur l'ensemble des catégories de la grammaire.

Thompson (1981, 1982, 1983) propose également un formalisme intéressant que nous comptons explorer. On peut aussi utiliser un agenda des hypothèses, règles, chemins à essayer. Dans un parseur GSG, cet agenda prendra la forme d'une liste d'hypothèses (catégories recherchées) ordonnée selon divers principes généraux. Ces mécanismes de contrôle pourront faire toute la différence entre un analyseur totalement inefficace et un autre, plus parcimonieux dans son travail et donc plus efficace et/ou plus rapide.

Les «charts», proposées par Kay (1970, 1973) et Kaplan (1973) (bien présentées dans Winograd 1983) constituent une structure de contrôle très utile et très répandue. On trouvera un bon exemple de l'utilisation de «charts» dans Martin et al. (1981), un analyseur RTA en largeur couvrant un sous-ensemble non négligeable de l'anglais.

Il pourrait se trouver que d'autres structures de contrôle soient mieux adaptées aux propriétés syntaxiques spécifiques des langues naturelles. Le cadre théorique de la grammaire syntagmatique généralisée, très riche bien que restrictif, permet d'envisager un certain nombre d'hypothèses à ce sujet.

4.7 *Représentation des données grammaticales*

Un dernier aspect des choix stratégiques à opérer dans l'analyse des langues naturelles, qui est certainement le plus important pour nous, en tant que linguistes s'intéressant aux propriétés générales du langage, concerne la représentation des données grammaticales.

Il y a d'abord le problème de la représentation des résultats de l'analyse, résultats intermédiaires, provisoires ou définitifs et résultats finals, structures de travail, etc. Mais le plus important concerne la nature des grammaires à utiliser, leur forme et leur fonctionnement en tant que modèles de compétence aussi bien que leur représentation dans l'analyseur (modèle de performance). On s'intéressera plus particulièrement à la représentation de l'information syntaxique (propriétés syntaxiques) et lexicale : forme et fonctionnement des règles, des entrées lexicales, clés de recherche dans celles-ci ou celles-là. Ce problème est à peine effleuré par Winograd (1983), qui semble présumer que son langage descriptif (DL) et les RTA qu'il présente sont optimaux de ce point de vue. En fait, je ne connais aucun travail, depuis Hays (1967), qui attaque directement ces questions essentielles.

C'est à ce niveau que le modèle de la grammaire syntagmatique généralisée apporte les solutions les plus intéressantes. Vu les limites d'espace, il ne nous est pas possible de faire une présentation globale de GSG.[8] On peut cependant noter les points suivants.

4.7.1 S-structure

En ce qui concerne la structure syntagmatique ou S-structure, GSG permet de rendre compte de chacun de ses aspects (catégories, ordre linéaire et hiérarchie) de façon autonome. En effet, elle contient une théorie généralisée des catégories, où les catégories sont des objets complexes avec une structure interne. Cette théorie possède un modèle formel, un langage, la théorie des traits.[9] De plus, le formalisme DI/PL (dominance immédiate / préséance linéaire) permet de formuler des généralisations sur chacune de ces propriétés : hiérarchie (e.g. métarègles), ordre linéaire (e.g. hypothèse d'ordre exhaustif partiel complet) en utilisant le vocabulaire très riche de la théorie des traits (e.g. métarègles, instanciations, etc.).

8. Cf. Gazdar (1982), Gazdar-Pullum (1982), Evans & Gazdar (1984), Morin (1984) et les références qui y sont citées.

9. Cf. Gazdar-Pullum (1982), Evans & Gazdar (1984), Morin (1984).

Thompson (1981, 1982, 1983) a démontré, dans plusieurs travaux récents, comment les idées de base de GSG pouvaient être exploitées dans un parseur sophistiqué de l'anglais. Shieber (1983b) traite plus particulièrement du formalisme DI/PL (ID/LP, en anglais) et de son utilisation directe dans un parseur. Plusieurs chercheurs, entre autres à SRI travaillent sur un modèle d'analyseur basé sur GSG, qui utilise directement la métagrammaire sans synthétiser de grammaire-objet.

4.7.2 Gouvernement

On a mentionné comment les problèmes du gouvernement pouvait être abordés de façon très générale au moyen d'un certain nombre d'hypothèses fortes mais très simples (indexation des I aux X_I, S-structure «plate», etc.). Notons encore que la théorie généralisée des catégories permet d'exprimer les relations de sélection (présence/absence d'une catégorie dans un domaine) aussi bien que de rection (forme d'une catégorie dans un contexte) de façon uniforme, puisque chacune des ces propriétés est exprimable sous forme de traits généralisés.

4.7.3 Dépendances

Les dépendances sont exprimables au moyen des catégories cornées («slashed categories»). On remarquera que la théorie généralisée des catégories permet de rendre compte aussi bien des dépendances concernant les catégories vides (où la valeur du trait de corne est une catégorie vide), que d'autres types de dépendances globales, négligées en grammaire transformationnelle (e.g. les dislocations, où la valeur du trait de corne est une catégorie formée de traits d'accord, i.e. un pronom).

En ce qui concerne le parsage, un tel formalisme permet une utilisation maximale de la structure de contrôle d'agenda.

5. Épilogue

Dans ces quelques notes, nous avons tenté de faire voir l'intérêt d'une approche du type de celle de la théorie de la grammaire syntagmatique généralisée pour une théorie du parsage des langues naturelles. On a pu voir que nombre d'aspects de la théorie et de son modèle se prêtaient directement à une utilisation optimale dans un analyseur syntaxique. Cela est d'autant plus étonnant que les développements théoriques récents en GSG sont à peu près totalement indépendants des considérations de parsage. En fait, la

théorie syntagmatique généralisée ne tire pas plus son origine dans les recherches sur le traitement automatique des langues naturelles que la grammaire transformationnelle ne tire la sienne des recherches en traduction automatique, bien qu'une certaine confusion se soit installée chez les opposants à ces deux théories. Confusion qui est d'ailleurs utilisée comme argument contre celles-ci (il s'agit d'un argument d'«impureté théorique» qui, dans un cas comme dans l'autre, est tout à fait non fondé).[10]

Ajoutons enfin que le cadre théorique de la grammaire syntagmatique généralisée est en évolution constante et a déjà suscité des variantes pas toujours compatibles (e.g. avec/sans métarègles, S-structure «plate» ou non, correspondance règle à règle, indexation, etc.) et des applications de taille appréciable :

Parseurs syntaxiques[11]

PSG
Linguistics Research Center
University of Texas
(cf. Bear 1982, Bear-Karttunen 1979)

PATR-I et *PATR-II*
SRI International
(cf. Shieber 1983, 1984, Shieber et alii 1983, Stucky 1983, Uszkoreit 1983)

MCHART
Department of Artificial Intelligence
University of Edinburgh
(cf. Thompson 1981, 1982, 1983)

Parseur sémantique

LM-GPSG
Computer Science Laboratories

10. À la décharge de ceux qui confondent la grammaire syntagmatique généralisée avec ses applications dans le traitement automatique des langues naturelles, il faut noter que cette théorie a effectivement des applications pratiques dans le domaine, bien qu'elle en soit tout à fait indépendante. Ce qui est loin d'être le cas en grammaire transformationnelle relativement à la traduction automatique.

11. Cf. aussi Heidorn (1975), un précurseur de la GSG. Dans ce domaine comme dans beaucoup d'autres, les applications précèdent la théorie qui devrait les sous-tendre : Nécessité fait loi.

Hewlett Packard Company
(cf. Gawron et alii 1982)

Système de traduction automatique
METAL
Linguistics Research Center
University of Texas
(cf. Slocum 1984)

Environnement d'édition-expérimentation GSG
ProGram
Cognitive Studies Programme
University of Sussex
(cf. Evans & Gazdar 1984)

C'est là, croyons-nous, un excellent indicateur du degré de fertilité d'une théorie.

Bibliographie

BAKER, C. (1979) «Syntactic theory and the projection problem», *Linguistic Inquiry*, n° 10, pp. 533-581.

BAKER, C. (1981 réd.) *The Logical Problem of Language Acquisition*, MIT Press.

BALLY, Ch. (1944) *Linguistique générale et linguistique française*, Francke.

BARLOW, M. et alii (1982) *Developments in GPSG, IULC.*

BATES, M. (1978) «The Theory and Practice of Augmented Transition Networks», dans Bolc, L. (1978 réd.) *Natural Language Communication with Computers*, pp. 191-260, Springer.

BEAR, J. (1982) «Gaps as Syntactic Features», *IULC.*

BEAR, J. et L. Karttunen (1979) «PSG : A Simple Phrase Structure Parser», *Texas Linguistic Forum*, n° 15, pp. 1-46.

BERWICK, R. (1982) «Computational Complexity and Lexical-Functional Grammar», *AJCL*, vol. 8, n° 3-4, pp. 97-108.

BERWICK, R. (1983) «Syntactic Constraints and Efficient Parsability», *ACL-21*, pp. 119-122.

BERWICK, R. et A. Weinberg (1982) «Parsing Efficiency, Computational Complexity and the Evaluation of Grammatical Theories», *LI*, vol. 13, n° 1, pp. 165-191.

BERWICK, R. et A. Weinberg (1983) «The Role of Grammars in Models of Language Use», *Cognition*, vol. 13, n° 1, pp. 1-61.

BOGURAEV, B.K. (1979) *Automatic Resolution of Linguistic Ambiguities*, thèse de Ph.D., TR-11, Computer Laboratory, Univ. of Cambridge.

BRESNAN, J. (1983 réd.) *The Mental Representation of Grammatical Relations*, MIT Press.

BURTON, R. (1976) «Semantic Grammar: An Engineering Technique for Constructing Natural Language Understanding Systems», BBN, TR-3453, Bolt, Beranek & Newman.

BURTON, R. et W.A. Woods (1976) «A Compiling System for Augmented Transition Networks», *COLING-76.*

CARROLL, J.A. (1982) «An Island Parsing Interpreter for Augmented Transition Networks», TR-33, Computer Laboratory, University of Cambridge.

CHESTER, D. (1980) «A Parsing Algorithm that Extends Phrases», *AJCL* vol. 6, n° 2, pp.87-96.

CHOMSKY, N. (1980) *Rules and Representations*, Blackwell.

CHURCH, K. (1982) «On Memory Limitations in Natural Language Processing», *IULC.*

CHURCH et R. Patil (1982) «Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table», *AJCL*, vol. 8, n^{os} 3-4, pp. 139-149.

COLMERAUER, A. (1970) «Les systèmes-Q ou un formalisme pour analyser et synthétiser des phrases sur ordinateur», PI-43, Département d'Informatique et de recherche opérationnelle, U. de M.

COLMERAUER, A. (1975) «Les grammaires de métamorphose», GIA, Univ. de Marseille-Luminy, (version anglaise dans Bolc 1978 réd).

COLMERAUER, A. (1982) «An Interesting Natural Language Subset», in Clark, K. et S.A. Tarnlund (1982 réd.), *Logic Programming*, Academic Press.

COWPER, E. (1976) *Constraints on Sentence Complexity: A Model for Syntactic Processing*, thèse de Ph.D., Brown University.

CRAIN et Fodor (1984) «How can grammars help parsers?», in Dowty et al. (1984 réd.).

DOWTY, D., L. Karttunnen et A. Zwicky (1984 réd.) *Natural Language Processing: Psycholinguistic, Computational and Theoretic Perspectives*, Cambridge Univ. Press.

ENGDAHL, E. (1982a) «Constituent questions, topicalization and surface structure semantic interpretation» dans Flickinger, D. et al. (1982 réd.) *Proceedings of the First West Coast Conference on Formal Linguistics* pp. 256-267, Stanford.

ENGDAHL, E. (1982b) «A Note on the Use of Lambda-Conversion in Generalized Phrase Structure Grammars» *Linguistics and Philosophy*, n° 4, pp. 505-515.

EVANS, R. & G. Gazdar (1984) *The ProGram Manual*, CSRP-035, Cognitive Studies Programme, University of Sussex.

FININ, T. et G. Hadden (1979) «Augmenting ATNS», *IJCAI-6*.

FODOR, J.D. (1978) «Parsing strategies and constraints on transformations», *LI*, n° 9 pp. 427-473.

FODOR, J.D. (1983) «Phrase Structure Parsing and the Island Constraints, *Linguistics and Philosophy*, n° 6 pp. 163-223.

FODOR, J.D. (1984) «Learnability and parsability: a reply to Culicover», *NLLT*, n° 2 pp. 105-150.

FODOR, J.D. et L. Frazier (1980) «Is the human sentence parsing mechanism an ATN?», *Cognition*, n° 8 pp. 417-459.

FORD, M., J. Bresnan et R. Kaplan (1983) «A competence-based theory of syntactic closure», dans Bresnan (1983 réd.).

FRAZIER, L. (1979) *On Comprehending Sentences: Syntactic Parsing Strategies, IULC.*

FRAZIER, L. et J.D. Fodor (1978) «The Sausage-machine: a new two-stage parsing model», *Cognition*, n° 6 pp. 291-325.

GAWRON, J.M. et alii (1982) «The GPSG Linguistic System», *ACL-20* : 74-81 et TN-CSL-82-5, Hewlett-Packard Labs.

GAZDAR, G. (1980) «A phrase-structure syntax for comparative clauses», in Hoekstra, T. et al. (1980 réd.) *Lexical Grammar*, Foris.

GAZDAR, G. (1982) «Phrase Structure Grammar» dans Jacobson, P. et G. Pullum (1982 réd.), *The Nature of Syntactic Representation*, Reidel.

GAZDAR, G. (1983) «Phrase Structure Grammars and Natural Languages», *IJCAI-8*, n° 1 pp. 556-565.

GAZDAR, G. (1984) «Recent Computer Implementations of Phrase Structure Grammar», University of Sussex, CSRP 024.

GAZDAR, G. et G. Pullum (1982) «GPSG: A Theoretical Synopsis», *IULC*.

GAZDAR, G., G. Pullum et I. Sag (1982) «Auxiliaries and related phenomena in a restrictive theory of grammar», *Language*, vol. 58 pp. 591-638.

GAZDAR, G. et al. (1982) «Coordinate structure and unbounded dependencies», in Barlow et al. (1982 réd.).

GAZDAR, G. et al. (1984 réd.) *Order, Concord and Constituency*, Foris.

HADDEN, G. (1977) «NETEDI: An Augmented Transition Network Editor», mémoire de M.Sc., Dept. of Computer Science, Univ. of Illinois.

HAYS, D.G. (1966 réd.) *Readings in Automatic Language Processing*, American Elsevier.

HAYS, D.G. (1967) *Introduction to Computational Linguistics*, American Elsevier.

HEIDORN, G. (1975) «Augmented Phrase Structure Grammars», *TINLAP-1* pp. 1-5.

HEIDORN, G. (1976) «Automatic programming through natural language dialogue: A survey», *IBM Journal of Research and Development*, vol. 20 p. 4.

HIRAKAWA, H. (1983) «Chart Parsing in Concurrent Prolog», *ICOT*, TR-008.

HUOT-LEMONNIER, F. (1984) *Études sur la syntaxe écrite des enfants du primaire*, thèse de Ph.D., Département de linguistique et philologie, Université de Montréal.

JOSHI, A. (1982) «Phrase Structure Trees bear more Fruit than you would have thought», *AJCL*, n° 8 pp. 1-11.

KAPLAN, A. (1950) «An experimental study of ambiguity and context», miméo, RAND Corporation.

KAPLAN, R. (1973) «A General Syntactic Processor», in Rustin (1973 réd.) pp. 193-241.

KAPLAN, R. (1975) *Transient Processing Load in Relative Clauses*, thèse de Ph.D., Harvard.

KARTTUNEN, L. (1981) «Unbounded Dependencies: Slash Categories vs. Dotted Lines», Univ. of Amsterdam, Mathematical Centre, Tract 135, pp. 323-342.

KAY, M. (1967) «Experiments with a Powerful Parser», *COLING-2*.

KAY, M. (1973) «The MIND System» in Rustin (1973) pp. 155-188.

KAY, M. (1977) «Morphological and Syntactic Analysis», dans Zampolli, A. (1977 réd.) *Linguistic Structures Processing* pp. 131-234, North-Holland.

KIMBALL, J. (1973) «Seven principles of surface structure parsing in natural language», *Cognition*, vol. 2, n° 1 pp. 15-47.

KIMBALL, J. (1982) «When Metarules are not Metarules», dans Barlow, M. et alii (1982 réd.) *Developments in GPSG, IULC*, pp. 72-94.

KITTREDGE, R. et J. Lehrberger (1981 réd.) *Sublanguages*, Walter de Gruyter.

KLEIN, E. (1982) «The interpretation of adjectival comparatives», *Journal of Linguistics*, vol. 18, pp. 113-136.

KOBAYASHI, Y. et Y. Niimi (1979) «A Procedural Representation of Lexical Entries in Augmented Transition Network Grammar», *IJCAI-6*.

KONOLIGE, K. (1980) «Capturing Linguistic Generalizations with Metarules in an Annotated Phrase-Structure Grammar», *ACL-18*, pp. 43-48.

KOOIJ, J.G. (1971) *Ambiguity in Natural Language*, North-Holland.

KUNO, S. (1966) «The Predictive Analyzer and a Path Elimination Technique», in Hays (1966 réd.), pp. 83-106.

KUNO, S. et A. Oettinger (1963) «Syntactic structure and ambiguity in English», *AFIPS Conference Proceedings*, vol. 24, pp. 397-418.

LAVOREL, P.M. (1979) «Grammaires pour les analyseurs morphosyntaxiques : Rappels théoriques et recettes pratiques», *T.A. Informations*, vol. 20, n° 1, pp. 3-24.

LIGHTFOOT, D. (1979) *Principles of Diachronic Syntax*, Cambridge University Press.

LYONS, J. (1968) *Theoretical Linguistics*, Cambridge Univ. Press.

LYONS, J. (1977) *Semantics*, 2 vol., Cambridge Univ. Press.

MARCUS, M. (1974) «Wait-and-See Strategies for Parsing Natural Language», MIT, AI-WP-36.

MARCUS, M. (1975) «Diagnosis as a Notion of Grammar», *TINLAP-1*.

MARCUS, M. (1976) «A Design for a Parser for English», *ACM-76*, pp. 62-68.

MARCUS, M. (1978a) «A Theory of Syntactic Recognition for Natural Language», thèse de Ph.D., MIT.

MARCUS, M. (1978b) «A Computational Account of Some Constraints on Language», *TINLAP-2*, pp. 236-246, aussi in Joshi, A., B. Webber et I. Sag (1981 réd.) *Elements of Discourse Understanding* pp. 177-200, Cambridge Univ. Press.

MARCUS, M. (1978c) «Capturing Linguistic Generalizations in a Parser for English», *2nd National Conference, Canadian Society for Computational Studies of Intelligence* pp. 19-21, pp. 64-73.

MARCUS, M. (1980) *A Theory of Syntactic Recognition for Natural Language*, MIT Press.

MARCUS, M., D. Hindle et M. Fleck (1983) «Talking about Talking about Trees», *ACL-21*, pp. 129-136.

MARTIN, W., K. Church et R. Patil (1981) «Preliminary Analysis of a Breadth-First Parsing Algorithm: Theoretical and Experimental Results», MIT, LCS, TR-261.

McCORD, M. (1980) «Slot Grammars», *AJCL*, vol. 6, n° 1, pp. 31-43.

McCORD, M. (1982) «Using Slots and Modifiers in Logic Grammars for Natural Language», *Artificial Intelligence*, vol. 18, n° 3, pp. 327-367.

MILNE, R. (1983) «An explanation for minimal attachment and right association», *ACL-21*, pp. 88-90.

MORIN, J.Y (1984) «Théorie syntagmatique généralisée» et «Théorie formelle des traits», dans Morin, J.Y. (1984) *Syntaxe*, Département de Linguistique et Philologie, U. de M., Appendices A et B, pp. 56-64 et 65-68.

PEREIRA, F. (1981) «Extraposition Grammars», *AJCL*, vol. 7, n° 4, pp. 243-256.

PEREIRA, F. (1982) «Logic for Natural Language Analysis», thèse de Ph.D., Univ. of Edinburgh.

PEREIRA, F. et D.H. Warren (1980) «Definite Clause Grammars for Language Analysis: A Survey of the Formalism and a Comparison with Augmented Transition Networks», *Artificial Intelligence*, vol. 13, pp. 231-278.

PEREIRA, F. et D.H. Warren (1983) «Parsing as Deduction», *ACL-21*, pp. 137-144.

PLANTE, P. (1981) *Déredec, Logiciel pour le traitement linguistique et l'analyse de contenu de textes*, Manuel de l'usager, Service de l'Informatique, Université du Québec à Montréal (4ième édition).

PRATT, V. (1975) «LINGOL — A progress report», *IJCAI-4*, pp. 422-428.

PULLUM, G. (1982) «Free Word Order and Phrase Structure Rules», *NELS-12*, pp. 209-220.

PULLUM, G. (1983) «Context-Freeness and the Computer Processing of Human Languages», *ACL-21*, pp. 1-6.

PULLUM, G. et G. Gazdar (1982) «Natural Languages and Context-Free Languages», *Linguistics and Philosophy*, n° 4, pp. 471-504.

PULMAN, S. (1982) «GPSG, Earley's Algorithm and the Minimisation of Recursion», dans Sparck-Jones, K. et Y. Wilks (1984 réd.), *Automatic Natural Language Parsing*, Academic Press.

RITCHIE, G. (1980) *Computational Grammar*, Harvester.

ROSS, K. (1980) «Parsing English Phrase Structure», thèse de Ph.D., Univ. of Mass., Amherst.

RUSTIN, R. (1973 réd.) *Natural Language Processing* Algorithmics Press.

SAGER, N. (1973) «The string parser for scientific literature», in Rustin (1983 réd.).

SAGER, N. (1981) *Natural Language Information Processing*, Addison-Wesley.

SALKOFF, M. (1973) *Une grammaire en chaîne du français*, Dunod.

SHEIL, B. (1976) «Observations on Context Free Parsing», *SMIL*, pp. 71-109.

SHIEBER, S. (1983) «Sentence Disambiguation by a Shift-Reduce Parsing Technique» *ACL-21*, pp. 113-118.

SHIEBER, S. (1984) «Direct Parsing of ID/LP Grammars», TN-291, *SRI* et *Linguistics and Philosophy*, n° 7, pp. 135-154.

SHIEBER, S. et alii (1983) «Formal Constraints on Metarules», *ACL-21*, pp.22-27.

SLOCUM, J. et alii (1984) «METAL: The LRC Machine Translation System», *ISSCO MT Symposium*, Lugano.

SMALL, S. (1979) «Word Expert Parsing», *ACL-17*.

SMALL, S. (1980) «Word Expert Parsing: A Theory of Distributed Word-Based Natural Language Understanding», thèse de Ph.D. Dept. of Computer Science, Univ. of Maryland.

SMALL, S., G. Cottrell et L. Shastri (1982) «Toward Connectionist Parsing», *AAAI*.

STOWE, L. (1984) *Models of Gap Location in the Human Language Processor, IULC*.

STUCKY, S. (1983) «Metarules as meta-node-admissibility conditions», TN-304, SRI International.

TENNANT, H. (1981) *Natural Language Processing*, Petrocelli.

TENNANT, H. et al. (1983) «Menu-based natural language understanding», *ACL-21*, pp. 151-158.

THOMPSON, H. (1981) «Chart Parsing and Rule Schemata in GPSG», *ACL-19*, pp. 167-172 et DAI Research Paper 165, Department of Artificial Intelligence, University of Edinburgh.

THOMPSON, H. (1982) «Handling Metarules in a Parser for GPSG» dans Barlow et alii (1982 réd.), pp. 26-37.

THOMPSON, H. (1983) «Crossed Serial Dependencies: A Low-Power Parseable Extension to GPSG», *ACL-21*, pp. 16-21.

USZKOREIT, H. (1983) «A Framework for Processing Partially Free Word Order», *ACL-21*, pp. 106-112.

VALIENT, L.G. (1975) «General Context-Free Recognition in Less than Cubic Time», *Journal of Computer and System Sciences*, vol. 10, pp. 308-315.

WALKER, D. (1978 réd.) *Understanding Spoken Language*, North-Holland.

WANNER, E. (1980) «The ANT and the Sausage-machine: Which one is baloney?», *Cognition*, n° 8, pp. 209-225.

WASOW, T., I. Sag et G. Nunberg (1982) «Idioms: an interim report», dans *Travaux préliminaires du XVIII^e congrès international des linguistes*, CIPL, Tokyo.

WEHRLI, E. (1983) «A GB Parser» *IJCAI-8*.

WEHRLI, E. (1984) «A government-binding parser for French», *ISSCO*, Université de Genève, WP-48.

WINOGRAD, T. (1972) *Understanding Natural Language*, Academic Press.

WINOGRAD, T. (1983) *Language as a Cognitive Process: Syntax* Addison-Wesley.

WOODS, W.A. (1970) «Transition Network Grammars for Natural Language Analysis», *CACM*, vol. 13, n° 10, pp. 519-606.

WOODS, W.A. (1973) «An Experimental Parsing System for Transition Network Grammars» in Rustin, R. (1973 réd.).

WOODS, W.A. (1980) «Cascaded ATN Grammars», *AJCL*, vol. 6, n° 1, pp. 1-12.

WOODS, W.A. et al. (1972) *The Lunar Sciences Natural Language Information System: Final Report (n° 2378), BBN.*