Deux problèmes d'analyse syntaxique automatique

Eric Wehrli
Université de Genève

1 Introduction

L'analyse syntaxique automatique des langues naturelles (ou *parsing*) est une étape fondamentale dans le processus d'analyse automatique du langage. C'est en effet à l'analyseur qu'incombe la double tâche d'une part de déterminer la catégorie grammaticale des éléments lexicaux du texte d'entrée, et d'autre part de tenter de dégager les structures syntaxiques des phrases de ce texte. Ce sont ces structures, ensuite, qui vont permettre de calculer les diverses interprétations sémantiques et pragmatiques.

Il n'est donc pas surprenant que le problème de l'analyse syntaxique automatique ait été l'objet d'une attention toute particulière, et ceci dès les premiers balbutiements des systèmes de traitement automatique des langues naturelles, mais surtout au cours des dix ou quinze dernières années. Cependant, bien que des progrès considérables aient été accomplis, il faut bien reconnaître que l'on est loin encore de disposer de modèles d'analyse suffisamment puissants et suffisamment souples pour satisfaire aux exigences d'applications à large échelle, telles que la traduction automatique ou le codage automatique de textes scientifiques (*automatic information encoding*). D'autre part, il convient également de souligner que les analyseurs développés jusqu'ici se révèlent peu satisfaisants d'un point de vue psycholinguistique, c'est-à-dire comme modèles de l'analyseur humain.

Dans cet article, j'aimerais aborder deux problèmes importants dans le développement des analyseurs syntaxiques. Il s'agit d'une

part du problème bien connu de l'ambiguïté, et d'autre part de celui de la détermination d'une stratégie d'analyse "réaliste", compatible avec les données psycholinguistiques sur les difficultés de traitement (*processing difficulties*).

Avant d'entamer la discussion de ces deux problèmes, il n'est peut-être pas inutile de rappeler brièvement ce que nous entendons par analyseur syntaxique.

2 Un bref survol

Comme l'illustre la figure 1 ci-dessous, un analyseur peut être considéré comme un mécanisme qui assigne à un texte d'entrée (limité ici à une phrase) un ensemble de représentations formelles – par exemple, des structures de constituants – comportant toutes les informations grammaticales et lexicales relatives à la phrase d'entrée.

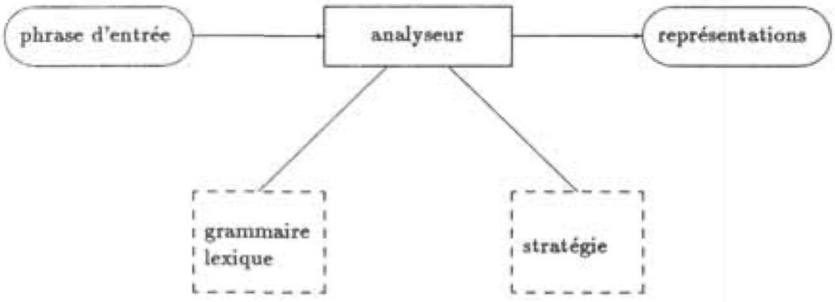

Figure 1: Structure d'un analyseur syntaxique

Pour donner un exemple concret, un analyseur syntaxique associera à une phrase comme (1a) une structure de constituants comme celle donnée en version très simplifiée en (1b):

(1)a. Le chat a mangé la souris.

b. [$_{S}$ [$_{NP}$ le chat] [$_{VP}$ a mangé [$_{NP}$ la souris]]]

Un analyseur consiste en deux parties bien distinctes: d'une part il doit avoir accès à certaines connaissances linguistiques, comprenant au moins la grammaire et le vocabulaire de la langue concernée, et d'autre part, il est doté d'une certaine stratégie, c'est-à-dire d'un algorithme qui spécifie dans leurs moindres détails toutes les étapes et toutes les opérations impliquées dans le processus d'analyse. En particulier, la stratégie décrit comment l'analyseur utilise ses connaissances linguistiques pour déterminer quelles sont les structures qui peuvent être assignées à la phrase qui lui est soumise. Il est commode de distinguer ces deux tâches dans le développement d'un analyseur. On parlera donc de spécification de la grammaire et du lexique utilisés pour l'analyse, ainsi que de spécification de la stratégie (ou des stratégies).

3 Le problème de l'ambiguïté

Le problème le plus évident sur lequel butent tous les analyseurs syntaxiques est celui de l'ambiguïté des langues naturelles. On le sait, pratiquement n'importe quelle phrase fourmille d'ambiguïtés, que ce soit au niveau lexical, syntaxique, sémantique ou pragmatique. L'effet conjugué de ces ambiguïtés peut avoir des conséquences désastreuses pour les analyseurs, aussi bien point de vue de leurs performances, qui décroissent en fonction du nombre de cas à considérer, que du point de vue de leurs résultats, puisqu'un analyseur qui assigne des dizaines de structures différentes à une phrase courante n'est guère utilisable[1].

A titre d'illustration, rien qu'au niveau lexical il est très facile de trouver des ambiguïtés. En effet, dans les langues naturelles, la plupart des mots peuvent être associés à plus d'un sens comme dans (2), voire à plus d'une catégorie syntaxique, comme dans (3):

(2)a. casse - action de casser
- cambriolage

[1] Il n'est pas rare, en effet, de trouver dans la littérature des exemples d'analyseurs qui attribuent à une phrase relativement simple, du moins pour un être humain, plusieurs dizaines, voire plusieurs centaines d'interprétations différentes. Cf. Robinson (1982), parmi beaucoup d'autres.

- casier contenant les caractères d'imprimerie
- ...

b. bande
- morceau d'étoffe
- film, pellicule, cassette
- partie étroite et allongée
- rebord de tapis de billard
- spectre (math.)
- fréquence (radio)
- groupe
- côté (maritime)
- ...

(3)a. ferme
- adjectif
- verbe
- substantif

b. force
- substantif
- verbe
- adverbe

c. est
- substantif
- verbe

d. passager
- substantif
- adjectif

On notera que, dans une très grande majorité, ces ambiguïtés n'affectent pas la communication. En fait, nous ne sommes en général pas conscients de ces ambiguïtés, qui passent donc inaperçues dans la communication quotidienne. Notre expérience de la communication ainsi que nos connaissances linguistiques et extralinguistiques font que dans un contexte donné, nous nous fixons très rapidement sur une interprétation (celle qui nous paraît sans doute la plus plausible à ce moment) et avons tendance à ignorer les autres.

S'il est vrai, comme on l'admet généralement aujourd'hui, que la solution globale du problème de l'ambiguïté des langues naturelles passe inéluctablement par l'interaction complexe de divers types de connaissances – y compris de connaissances non-linguistiques pour lesquelles on ne dispose pour l'instant que de modèles très limités – il

n'en reste pas moins que des solutions locales, c'est-à-dire propres à une composante du système, doivent également être développées. Les diverses composantes d'un système de traitement du langage doivent être capables de gérer aussi efficacement que possible les ambiguïtés qu'elles ne sont pas capables d'éliminer. Or, comme nous allons le montrer dans les paragraphes suivants, les méthodes généralement utilisées dans le développement des analyseurs syntaxiques ne semblent pas aptes à assurer cette tâche.

Pratiquement tous les analyseurs développés à ce jour reposent essentiellement sur des grammaires syntagmatiques. Quant aux stratégies d'analyse, elles s'inspirent largement des techniques d'analyse utilisées pour les langages artificiels, comme les langages de programmation. Très performantes pour des langages non-ambigus, ces techniques se révèlent vite inefficaces lorsqu'on les applique à des langages aussi ambigus que les langues naturelles[2]. En effet, toute tentative d'étendre la couverture syntaxique de l'analyseur – c'est-à-dire d'augmenter le nombre de constructions syntaxiques que l'analyseur est capable de reconnaître – ou d'améliorer la finesse des analyses, conduit inéluctablement à une augmentation des règles de grammaire et/ou des éléments terminaux homographes. Cette croissance de la grammaire et du lexique entraîne une multiplication rapide du nombre de combinaisons que l'analyseur doit considérer, ce qui, comme nous l'avons vu, conduit à une dégradation rapide des performances de l'analyseur.

Il est important de noter que le problème n'est pas exclusivement computationnel, puisque parallèlement à la prolifération des possibilités que l'analyseur doit prendre en considération, on assiste également à une prolifération des résultats. L'analyseur fournit ainsi plusieurs analyses (parfois un très grand nombre d'analyses), là où l'analyseur humain ne voit peut-être aucune ambiguïté (*cf.* note 1). La solution habituelle à ce problème, qui consiste à augmenter et à renforcer les conditions sur l'application des règles de la grammaire de façon à bloquer un plus grand nombre de dérivations, se révèle être à double tranchant: s'il est vrai que l'on peut, dans une cer-

[2]Voir Aho & Ullmann (1972) pour une discussion détaillée de divers algorithmes d'analyse, King (1983) et Winograd (1983) pour un panorama des problèmes spécifiques de l'analyse syntaxique automatique des langues naturelles.

taine mesure, filtrer les analyses de façon à éviter de trop nombreuses réponses, cette solution a l'inconvénient de rendre le système de plus en plus restrictif, à tel point qu'un grand nombre de phrases soumises à l'analyseur risquent d'être rejetées purement et simplement parce qu'une ou plusieurs des conditions ne sont pas satisfaites.

3.1 Ambiguïtés et modularité

Outre les tentatives très idéalistes des analyseurs déterministes[3], des solutions partielles au problème de la multiplicité des structures ont été proposées, basées sur l'idée du partage de structure (*structure sharing*)[4]. Ces solutions tirent parti du fait que les nombreuses structures qu'un analyseur est amené à considérer sont très souvent partiellement redondantes, en ce sens qu'elles ne diffèrent que par certaines de leurs sous-structures, toutes les autres étant rigoureusement identiques. L'idée est donc d'éviter la multiplication des sous-structures communes grâce au partage.

Il est clair que de telles solutions ne sont utiles dans la pratique que si le nombre de redondances partielles est élevé. Le raffinement des analyses par l'adjonction de marqueurs sémantiques ou pragmatiques pour distinguer des lectures marginalement différentes tend à réduire ces redondances et, par conséquent, restreint considérablement les avantages théoriques du partage de structure.

Pour prendre un exemple simple, face à des verbes très ambigus, comme *donner* et *faire*, les analyseurs traditionnels auront tendance à multiplier le nombre d'analyses par le nombre de lectures possibles pour ces verbes. En d'autres termes, ces analyseurs vont considérer les différentes lectures d'un verbe comme autant d'éléments distincts, même s'ils partagent un environnement syntaxique semblable. Il s'ensuit que tous les attachements envisagés –de spécificateurs, d'ajouts ou de compléments– devront être répétés pour chacune des structures.

Le recours à une organisation modulaire de l'analyseur basée sur une théorie syntaxique qui distingue plusieurs niveaux de représenta-

[3] Pour les analyseurs déterministes, voir Marcus (1980), Berwick et Weinberg (1984), Milne (1986).

[4] *Cf.* Kay (1980), Tomita (1986).

tion, comme c'est le cas par exemple de la théorie "gouvernement et liage" de Chomsky (1981, 1986), semble de nature à changer cet état de choses. Au lieu d'un niveau de représentation unique, où toute l'information disponible doit être représentée, une approche modulaire divise le problème de l'analyse syntaxique en sous-problèmes correspondant à des niveaux d'analyse distincts. L'information disponible à l'intérieur de chacun des modules est limitée au type d'information pertinente pour ce niveau d'analyse (*information encapsulation*). Cette factorisation de l'information conduit à une simplification notable des structures d'information, ce qui entraîne une augmentation de la fréquence des redondances partielles et par conséquent des possibilités de partage de structure[5].

Ainsi, un analyseur basé sur une conception modulaire de la grammaire se trouverait en principe dans une position plus avantageuse que ses concurrents basés sur des grammaires syntagmatiques pour éviter la prolifération des structures redondantes. Par exemple, un analyseur modulaire pourrait observer le principe suivant:

(4) *Principe d'unicité des structures:*
A l'intérieur d'un module, toutes les structures doivent être distinctes les unes des autres.

Le principe d'unicité exige que, dans un module donné, toutes les structures soient uniques, c'est-à-dire qu'elles soient toutes distinctes en termes de traits pertinents pour ce module. Toutes les structures qui ne diffèrent les unes des autres qu'en termes de traits non-pertinents sont regroupées et apparaissent comme un élément unique. Pour donner un exemple concret, dans le module X, responsable de l'assignation des structures de constituants, une structure unique est assignée au syntagme [$_{NP}$ *les fils*], même si celui-ci permet plus d'une interprétation. En effet, le fait que *fils* puisse être le pluriel de *fil* ou de *fils* n'est pas pertinent pour la construction du syntagme et est donc ignoré dans ce module. Il en va de même pour les exemples suivants:

(5)a. les faits, les voiles, *etc.*

[5]Pour une discussion des analyseurs basés sur la théorie "gouvernement et liage", voir Abney et Cole (1985), Barton (1984), Berwick (1987), Berwick et Weinberg (1984), Wehrli (1988).

b. a été reconnue, nous a fait, *etc*[6].

Le même scénario se produit à d'autres niveaux d'analyse. Ainsi, deux lectures qui ne diffèrent qu'en termes de fonctions thématiques partageront la même structure syntaxique, comme dans l'exemple suivant, où le syntagme *aux enfants* peut être interprété soit comme l'agent, soit comme le bénéficiaire du verbe *porter*.

(6) Jean a fait porter les livres aux enfants.

Ici, à nouveau, cette distinction ne joue aucun rôle du point de vue de la structure et, en vertu du principe d'unicité des structures, n'est donc pas pertinent à ce niveau. On observe que contrairement aux analyseurs classiques qui, comme les premiers modèles de la grammaire générative, tendent à ramener toutes les différences de sens à des différences de structures[7], un analyseur modulaire admet que des interprétations distinctes soient associées à une structure unique. Cet appauvrissement relatif de la représentation structurale ne s'accompagne pas de perte d'information puisque celle-ci est calculable au niveau de modules spécialisés, dans ce cas, le module responsable de l'assignation des fonctions thématiques. Du point de vue du traitement du langage, le fait de distinguer plusieurs niveaux d'analyse présente certains avantages. Cela permet d'une part de minimiser la masse des informations à considérer à un niveau particulier; d'autre part, dans la mesure où il existe un certain ordre d'application (ordre intrinsèque) des règles et des composantes, le filtrage à un niveau donné permet d'éviter toutes les opérations qui découlent de l'application des niveaux ultérieurs. Enfin, on s'attend à ce que cette modularisation de l'analyse permette d'éviter, ou tout au moins de minimiser, les effets combinatoires.

[6] Nous faisons ici l'hypothèse que les auxiliaires sont des spécificateurs du verbe.

[7] Qu'on se rappelle, par exemple, que dans le modèle transformationnel classique, l'ambiguïté d'un syntagme nominal comme *la peur du lion* dérive de deux structures distinctes, en gros, [$_S$ le lion a peur] et [$_S$ quelqu'un a peur du lion].

4 Le choix d'une stratégie

Un autre problème fondamental dans le développement d'analyseurs syntaxiques, problème qui se révèle crucial pour les analyseurs "réalistes", est celui du choix de la stratégie d'analyse. Par stratégie d'analyse, nous entendons l'algorithme, c'est-à-dire la spécification explicite de la séquence d'opérations élémentaires impliquées dans le traitement d'une phrase.

On comprend mieux la complexité du problème si l'on considère quelques-uns des paramètres dont il faudra tenir compte:

- analyse ascendante, descendante ou mixte, c'est-à-dire choix entre un analyseur gouverné par les données, par des prédictions ou par un mélange des deux
- traitement séquentiel ou parallèle des alternatives
- interaction de l'analyseur syntaxique avec les autres composantes de la grammaire (p.ex. sémantique)
- interaction de l'analyseur avec les composantes extralinguistiques (connaissance du monde, bon sens, etc.)

En fait, même la relation entre la grammaire et l'analyseur n'est pas aussi simple qu'on pourrait le penser. Il semble bien, en effet, qu'il faille écarter l'hypothèse d'une coïncidence totale entre l'espace de grammaticalité, défini comme l'ensemble de toutes les phrases grammaticales, et l'espace d'interprétabilité syntaxique, correspondant à l'ensemble de toutes les phrases interprétables syntaxiquement. Contrairement aux langages artificiels, où toute phrase grammaticale est du même coup assurée d'être interprétable syntaxiquement, cette équivalence ne vaut pas pour le langage naturel. Ainsi, il est facile de montrer que de nombreuses phrases qui ne respectent pas certaines règles de la grammaire sont néanmoins parfaitement analysables syntaxiquement.

S'il est vrai que certaines phrases grammaticalement malformées restent interprétables syntaxiquement, la situation contraire existe également, où certaines phrases grammaticalement bien formées ne

sont pas interprétables syntaxiquement. Ainsi, on a répértorié certains types de phrases parfaitement grammaticales mais dont la lecture peut poser des problèmes parfois insurmontables. Nous considérerons à tour de rôle les deux principales classes d'exemples discutées dans la littérature psycholinguistique récente, soit les limites liées à l'auto-enchâssement et les phrases-labyrinthes[8].

4.1 Le phénomène d'auto-enchâssement (*center-embedding*)

Chomsky et Miller (1963) sont à notre connaissance les premiers a avoir mis en lumière un conflit très intéressant entre grammaires et interpréteurs. Si la nécessité de règles récursives dans la grammaire semble indiscutable, on constate qu'au niveau de l'interprétation, il existe en fait une limite très stricte sur la profondeur des structures auto-enchâssées que nous sommes capables d'interpréter[9]. Alors qu'un niveau unique d'auto-enchâssement est parfaitement compréhensible, comme l'illustre l'exemple (7), un double niveau d'enchâssement devient pratiquement impossible à interpréter, ainsi que l'attestent les exemples (8), où l'astérisque signifie non pas l'agrammaticalité mais bien l'inacceptabilité d'une phrase.

(7) The cat the dog chased meowed.

(8)a. *The cat the dog the man likes chased meowed.

b. *The cat that the dog that the man likes chased meowed.

c. *That that that John left surprised us is amazing shocked the world.

Les jugements sont en général très consistants et semblent également valables pour le français:

(9) Le chat que le chien poursuit miaule.

(10)a. *Le chat que le chien que l'homme aime poursuit miaule.

[8] *Cf.* Ford, Bresnan et Kaplan (1983), Frazier et Fodor (1978), Pritchett (1988).

[9] On appelle auto-enchâssement le fait que l'on puisse engendrer à partir d'un certain symbole de la grammaire, disons A, une séquence de symboles contenant A avec un contexte gauche et un contexte droit non nuls, formellement: $B \stackrel{*}{\Longrightarrow} xBy$, *avec* $B \in V_N$, *et* $x,\ y \in V^+$.

b. *que que que Jean ait quitté la séance vous surprenne nous étonne stupéfie Marie.

L'intérêt de tels exemples pour le développement de stratégies d'analyse est évident. S'il est vrai que les grammaires de l'anglais et du français permettent la récursivité (et donc l'auto-enchâssement), le problème que pose l'interprétation des phrases ci-dessus ne peut être ramené à un principe de la grammaire, mais doit être expliqué en termes de stratégie d'analyse, ou peut-être en termes de limitations de la mémoire à court terme, comme le suggèrent d'ailleurs Chomsky et Miller.

4.2 Les phrases-labyrinthes (*garden-path sentences*)

Un autre exemple de conflit entre grammaires et analyseurs, encore mieux connu que le précédant, est celui des phrases-labyrinthes. A nouveau, ces phrases ont la particularité d'être pratiquement ininterprétables bien que parfaitement grammaticales. A leur lecture, on a tendance à s'engager sur une fausse piste, dont on a d'ailleurs souvent beacoup de mal à s'extraire, alors qu'il existe une alternative bien formée. Les phrases données en (11) sont les exemples classiques discutés dans la littérature psycholinguistique. La première traduction correspond, en gros, à l'analyse dans laquelle le lecteur se perd, alors que la deuxième traduction correspond à l'analyse grammaticalement bien formée mais qui échappe habituellement à la perspicacité du lecteur.

(11)a. The horse raced past the barn fell.
'le cheval s'élança au-delà de la grange tomba'
'le cheval qu'on a fait courir au-delà de la grange est tombé'

b. The boat floated quickly sank.
'le bateau flotta rapidement sombra'
'le bateau mis à flots sombra rapidement

Le phénomène de la phrase-labyrinthe semble lié à une double ambiguïté locale: il y a d'une part, l'ambiguïté morphologique du verbe *raced*, qui s'interpète soit comme la forme du passé, soit comme

le participe passé du verbe *to race*. La deuxième ambiguïté, elle, est de nature sélectionnelle. Des verbes commes *race* et *float* peuvent être utilisés soit transitivement soit intransitivement. Mises ensemble, ces deux ambiguïtés déterminent donc toute une classe d'exemples pour lesquels on aura le choix, localement, entre une lecture intransitive du verbe au temps du passé (*the horse raced, the boat floated*), et une lecture participiale du verbe, basée sur sa forme transitive et utilisée comme une relative réduite, lecture que l'on peut paraphraser comme suit:

(12)a. the horse that had been raced past the barn fell.

b. the boat that had been floated sank.

Typiquement, les lecteurs auxquels on soumet les phrases (11) s'engagent dans la première des deux interprétations (*raced* comme verbe principal), alors que seule la seconde interprétation permet d'obtenir une structure bien formée pour ces phrases.

Bien que cette double ambiguïté soit particulièrement fréquente en anglais, liée dans une large mesure à la fréquence de l'ambiguïté morphologique entre les formes verbales du passé et du participe passé, on peut également trouver des exemples de phrases-labyrinthes en français. Ainsi, toutes les personnes à qui nous avons fait lire les phrases données en (13) les ont rejetées comme innaceptables ou incompréhensibles:

(13)a. Le modèle qu'il a fait penser à un arbre de Noël.

b. Le pouvoir énorme qu'il a fait de lui l'homme le plus redouté de la province.

c. Un des amis qu'il a fait construire un château en Espagne.

Comme pour les exemples anglais discutés plus haut, les phrases (13) recèlent une ambiguïté locale qui n'est pas perçue par le lecteur. A nouveau, il y a double ambiguïté: d'une part l'ambiguïté du verbe *avoir*, qui peut être interprété soit comme verbe principal, soit comme auxiliaire, et celle de la forme verbale *fait*, qui correspond à la forme du participe passé du verbe *faire* de même qu'à une des formes du

présent de l'indicatif. Ce qu'on observe dans les phrases (13), c'est une tendance systématique à interpréter la séquence *a fait* comme une forme verbale composée (passé composé du verbe *faire*) plutôt que comme une séquence de deux verbes indépendants, l'un correspondant au verbe de la phrase relative, le second au verbe principal de la phrase. Or, le fait que ces phrases sont bien formées syntaxiquement est confirmé par les exemples suivants, dont la forme est tout à fait parallèle à celle des phrases (13) mais où l'ambiguïté d'une des deux formes verbales a été supprimée:

(14)a. Un des amis qu'il a possède un château.

b. Le premier modèle qu'il a eu faisait penser à un arbre de Noël.

Bien que manifestement très marginaux, ces phénomènes sont extrêmement précieux puisqu'ils révèlent certaines propriétés fondamentales de l'analyseur humain. Le travail des linguistes-informaticiens et des psycholinguistes consiste à tenter d'élaborer des modèles d'analyseurs syntaxiques capables de rendre compte de ces phénomènes et donc capables de reproduire le comportement de l'analyseur humain face à des constructions comme celles que nous venons de discuter.

5 Conclusion

Les deux problèmes abordés dans cette brève discussion illustrent bien l'écart qu'il y a entre l'analyse syntaxique des langages artificiels, telle qu'on la conçoit par exemple dans le développement de compilateurs pour les langages de programmation, et l'analyse syntaxique des langues naturelles. En effet, d'une façon générale, les langages artificiels ne connaissent ni le problème de l'ambiguïté, ni celui de la distinction entre grammaticalité et interprétabilité. Il s'ensuit, naturellement, que les solutions aux problèmes de l'analyse syntaxique dans ces deux types de langages doivent faire appel à des approches différentes. Ceci peut paraître évident, mais l'histoire du traitement automatique du langage montre que l'on a trop souvent sous-estimé la spécificité du langage naturel et cherché à lui

imposer des techniques d'analyse développées pour les langages artificiels. Il est temps de s'écarter de ces vues par trop réductionnistes du traitement des langues naturelles et de se tourner davantage vers la linguistique théorique et la psycholinguistique que vers les techniques de compilation pour l'élaboration des modèles de traitement automatique du langage.

6 Bibliographie

Abney, S. et J. Cole, 1985. "A government-binding parser", *NELS XVI.*

Aho, A. et J.D. Ullman, 1972. *The Theory of Parsing, Translation and Compiling*, Prentice-Hall.

Berwick, R., 1987. "Principle-based parsing", rapport technique, MIT AI-lab.

Berwick, R. et A. Weinberg, 1984. *The Grammatical Basis of Linguistic Performance: Language Use and Acquisition*, MIT Press.

Chomsky, N., 1981. *Lectures on Government and Binding*, Foris Publications.

Chomsky, N., 1986. *Knowledge of Language: Its Origin, Nature and Use*, Praeger Publishers, New York.

Chomsky, N. et G. Miller 1963. "Introduction to the formal analysis of natural language" in R. Luce, R. Bush and E. Galanter (éd.) *Handbook of Mathematical Psychology*, Wiley.

Ford, M., J. Bresnan et R. Kaplan, 1983. "A competence-based theory of syntactic closure" in J. Bresnan (éd.), *The Mental Representation of Grammatical Relations*, MIT Press.

Frazier, L. et J.D. Fodor, 1978. "The sausage machine: a new two-stage parsing model", *Cognition* 6, 291-325.

Kay, M., 1980. "Algorithm schemata and data structures in syntactic processing", rapport technique CSL-80-12, Xerox Palo Alto Research Center.

King, M. (éd.), 1983. *Natural Language Parsing*, Academic Press.

Marcus, M., 1980. *A Theory of Syntactic Recognition for Natural Language*, MIT Press.

Milne, R., 1986. "Resolving lexical ambiguity in a deterministic parser", *Computational Linguistics* 12.1, 1-12.

Pritchett, B., 1988. "Garden path phenomena and the grammatical basis of language processing", *Language* 64:3, 539-576.

Robinson, J., 1982. "Diagram: a grammar for dialogues," *Communications of the Association of computing Machinery* 25:1, 27-47.

Tomita, M., 1986. *Efficient parsing for natural language: A fast algorithm for practical systems*, Kluwer Academic Publisher.

Wehrli, E., 1988. "Parsing with a GB grammar," in U. Reyle and C. Rohrer (eds.), *Natural Language Parsing and Linguistic Theories*, Reidel, 1988.

Winograd, T., 1983. *Language as a Cognitive Process*, Addison-Wesley.