



**HAL**  
open science

# La linguistique au contact de l'informatique : de la construction des grammaires aux grammaires de construction

Marcel Cori, Jean-Marie Marandin

► **To cite this version:**

Marcel Cori, Jean-Marie Marandin. La linguistique au contact de l'informatique : de la construction des grammaires aux grammaires de construction. *Histoire Epistémologie Langage*, 2001, 23, pp.49-79. halshs-00086492

**HAL Id: halshs-00086492**

**<https://shs.hal.science/halshs-00086492v1>**

Submitted on 18 Jul 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## La linguistique au contact de l'informatique : de la construction des grammaires aux grammaires de construction<sup>1</sup>

### Marcel Cori

Université Paris X – Nanterre  
FRE CNRS « Modèles, dynamiques, corpus »  
UFR L L Phi – Bâtiment L  
200 avenue de la République  
92001 Nanterre Cedex  
[mcori@u-paris10.fr](mailto:mcori@u-paris10.fr)

### Jean-Marie Marandin

Laboratoire de Linguistique Formelle (CNRS)  
Université Paris 7 – Denis Diderot  
UFRL, case 7003  
2 place Jussieu  
75251 Paris Cedex 05  
[jean-marie.marandin@linguist.jussieu.fr](mailto:jean-marie.marandin@linguist.jussieu.fr)

La linguistique de la seconde moitié du vingtième siècle est profondément marquée par le programme de recherche défini sous le terme de grammaire générative. La grammaire générative définit un nouvel objet pour la linguistique : la compétence. Chomsky reprend le terme de *grammaire* pour désigner un modèle de compétence. La linguistique se définit dès lors comme une entreprise de construction de grammaires (*grammar design*) qui sont tout à la fois des instances de description des langues et des modèles de la compétence :

- (1) It is assumed that there is a correct generative grammar which determines the structural characteristic of any string, including those that deviate from well-formedness in various respects, and that the problem of the linguist is to discover this grammar and to discover the linguistic theory that determines possible grammars and incorporates universals of language.  
(Chomsky, 1975 : 44)

Durant la même période, et parfois dans les mêmes institutions, se développent un ensemble de recherches, que nous regroupons sous l'étiquette *informatique*, consacrées à la modélisation des activités de programmation d'un ordinateur. Ces recherches incluent la théorie de la compilation, les langages de programmation, l'intelligence artificielle et le traitement automatique des langues (TAL). Chacune à sa manière croise le chemin de la linguistique, que ce soit au travers des notions de langage (et des usages du langage), de compétence qui est mis au fondement du programme cognitiviste, ou moins spécifiquement de connaissance. Les langages de programmation doivent être décrits syntaxiquement, interprétés et traduits en langage machine. L'intelligence artificielle se présente comme la modélisation de toute forme de connaissance ; si la grammaire est une forme de connaissance, la grammaire est du ressort de l'intelligence artificielle. Le TAL (dont le statut est ambigu, entre théorie de la performance et applications industrielles) est directement confronté aux aspects systématiques des langues naturelles.

La linguistique et l'informatique sont des disciplines « en contact » comme on dit que deux langues sont en contact : ignorance réciproque, rivalité plus ou moins belliqueuse, emprunt et collaboration. Dans cet article, nous ne construirons qu'un seul fil de l'histoire de ce contact : celui qui va de l'émergence d'un sous-programme consacré aux grammaires formelles à l'émergence d'une redéfinition du programme génératif initial, les grammaires de constructions. Notre entreprise n'est pas une entreprise d'historiens (ou de sociologues) des sciences ; elle s'apparente davantage à une tentative d'histoire rationnelle (Lakatos, 1984) : nous reconstruisons dans l'épaisseur des faits

---

<sup>1</sup> Nous remercions J. Léon et O. Bonami de leurs relectures et suggestions.

institutionnels et intellectuels les évolutions d'un programme de recherche qui aboutissent à un point de vue bien particulier sur la grammaire des langues qui nous intéresse, car nous en sommes partie prenante. C'est dire que notre point de vue est « engagé », partiel et très nettement délimité. En particulier, nous laisserons de côté les avatars des grammaires transformationnelles (proposés dans les modèles Théorie standard, Principes et paramètres, Gouvernement et liage, Programme minimaliste) car ils ne nous semblent pas contribuer de façon décisive à l'avancement du projet défini par Chomsky que nous rappelons en (1). Ce choix présuppose que l'on peut distinguer dans les théories linguistiques les propositions et les argumentations qui relèvent de la description des langues et celles qui relèvent du choix et de la confection des grammaires.

Pour la clarté, nous reconstruisons trois moments. Le premier est un moment fondateur pour la linguistique et l'informatique : la définition d'une nouvelle discipline, appelée par Chomsky *linguistique algébrique*, consacrée à l'étude des grammaires formelles. De façon paradoxale, les premiers résultats de cette discipline, la hiérarchie des grammaires, sont à l'origine de développements considérables en informatique, aussi bien en théorie de la compilation que dans la définition d'algorithmes d'analyse qui peuvent être investis dans le TAL, alors qu'ils sont cantonnés par Chomsky à servir de base à la réfutation des formes non transformationnelles de grammaire. Le second est un moment de réforme, où Gazdar joue un rôle déterminant : le programme génératif se scinde en deux. Le rejet des grammaires transformationnelles par les linguistes tenants de la forme syntagmatique des grammaires et les informaticiens tenants de la déclarativité se présente comme un retour au programme initial, et en particulier comme l'affirmation de la pertinence de l'approche formelle de la grammaire des langues naturelles. Le troisième moment est contemporain ; on pourrait filer la métaphore historique et parler de moment baroque (alors que le programme minimaliste apparaît comme une entreprise de contre-réforme dans ses tenants sinon dans ses aboutissants formels). L'analyse des langues inclut de nouvelles dimensions à la suite du développement d'approches formalisées en sémantique et en pragmatique. On demande aux grammaires d'intégrer ce qui relève du signifiant, du signifié et de l'usage des expressions linguistiques. C'est ce que recouvre la notion générale de construction. Cela est rendu possible par le développement en informatique de systèmes de représentation des connaissances complexes et multidimensionnels. HPSG (Pollard et Sag) apparaît comme le catalyseur contemporain de cette nouvelle approche de la grammaire, où l'entreprise de description des langues prend le pas sur la modélisation de la faculté de langage.

## **1. L'émergence du paradigme génératif**

### **1.1 Le contexte**

Chomsky décrit longuement le contexte d'émergence du programme génératif dans l'introduction qu'il rédige en 1973 pour la publication en 1975 de *The logical structure of linguistic theory* (LSLT dorénavant) qui date de 1955 ; le passage (2) résume l'essentiel de cette description :

(2) Interdisciplinary approaches to language, communication, and human behavior were much in vogue -- thus it was hardly surprising that a student in Cambridge in the early 1950s should have come to think of linguistics as, in effect, a branch of cognitive psychology concerned with the language faculty and its exercise. Roman Jakobson's work was well known and influential. Oxford ordinary language analysis and Wittgenstein's later work were attracting great interest. [...]. Mathematical logic, in particular recursive function theory and metamathematics, were becoming more generally accessible, and developments in these areas seemed to provide tools for a more precise study of natural language as well. All of this I personally found most stimulating.

At the same time electronic computers were beginning to make their impact. The mathematical theory of communication, cybernetics, sound spectrography, psychophysics, and experimental psychology were in a period of rapid development and much exuberance. Their contribution

lent an aura of science and mathematics to the study of language and aroused much enthusiasm [...]. A technology of machine translation, automatic abstracting, and information retrieval was put forward as a practical prospect. It was widely believed that B.F. Skinner's William James lectures of 1947 offered an account of some of the most complex products of human intelligence in terms of the science of behavior [...]. My personal reaction to this particular complex of beliefs, interests, and expectations was almost wholly negative. The behaviorist framework seemed to me a dead end, if not an intellectual scandal. The models of language that were being discussed and investigated had little plausibility [...] and I had no personal interest in the experimental studies and technological advances. The latter seemed to me in some respect harmful in their impact [...]. As for machine translation and related enterprises, they seemed to me pointless as probably quite hopeless.  
(Chomsky, 1975 : 39)

Cette présentation est autant une reconstruction imaginaire qu'une description factuelle du contexte d'émergence du programme génératif. Néanmoins, elle restitue correctement le socle épistémologique de la notion clef pour le programme génératif et pour les rapports entre linguistique et informatique, la notion de grammaire. Et cela passe par une claire distinction entre d'un côté les mathématiques (« function theory and metamathematics ») et de l'autre l'informatique que Chomsky amalgame à l'ennemi qu'il donne au programme naissant : le behaviourisme. L'idée de grammaire prend forme dans le sillage des mathématiques symboliques et non dans les programmes ou les théories (même mathématisées) de la science de l'information naissante. En effet, pour définir ce qu'il entend par une grammaire, Chomsky se réfère à la notion de système constructionnel (*constructional system*), qui est ainsi au cœur du nouveau programme de recherche. La forme organisatrice d'une grammaire générative n'est pas le répertoire comme c'est le cas pour les grammaires traditionnelles<sup>2</sup>, mais la forme logique d'une théorie.

### 1.1.1. La notion de système constructionnel

Un système constructionnel (SC dorénavant) est, en toute généralité, la reconstruction logique d'un savoir sur une région du monde. Carnap en a dessiné l'idée, et un type particulier, dans *Der logische Aufbau der Welt*. C'est Goodman qui, sur la base d'une critique de Carnap, généralise la notion dans *Structure of Appearance* (1951 ; SA dorénavant). Les argumentations de Chomsky sont soutenues par les résultats auxquels est parvenu Goodman<sup>3</sup>.

Goodman insiste sur le fait que les éléments primitifs de toute description peuvent recevoir plusieurs définitions adéquates (*adequacy*) et que le choix d'un système de définitions dépend de la fonction qu'on assigne au système global, et en particulier du gain de précision (*accuracy*) que l'on pense en tirer. Goodman récuse donc avec force l'idée que la description d'une région du monde demande que soit isolé un ensemble unique et fixe d'éléments, en particulier d'éléments qui auraient le statut de données identifiées préalablement et indépendamment de la description elle-même<sup>4</sup>. Il découle de cette prémisse que tout savoir sur une région du monde donne lieu à plusieurs systèmes (selon la base empirique que l'on se donne au départ) et que le critère pour évaluer les systèmes en concurrence implique nécessairement une évaluation de l'effet de connaissance que chacun apporte.

Chomsky reprend intégralement ces propositions. On peut soutenir que ce sont ces propositions qui constituent l'essentiel de la rupture avec le programme linguistique contemporain, tel que l'ont

---

<sup>2</sup> Répertoire des expressions attestées, curieuses, grammaticales ou correctes selon les grammaires.

<sup>3</sup> Chomsky cite plusieurs fois SA dans l'introduction de LSLT ainsi que dans la première note de *Syntactic Structures*. C'est un indice faible étant donné le fait que le régime de citation et de non-citation chez Chomsky (et plus généralement dans le groupe qu'il contrôle) obéit plus à une logique institutionnelle qu'à la reconnaissance de dettes intellectuelles. La référence à « la structure logique de la théorie » est un indice bien plus significatif d'une filiation avec SA.

<sup>4</sup> « The consideration relevant in choosing elements for a system is thus not primacy in the cognitive process but serviceability as a basis for an economical, perspicuous and integrated system » (Goodman, 1972 : 10).

façonné les travaux de Bloomfield et Harris et auquel Chomsky a été exposé dans ses années de formation. La rupture ne porte pas sur le contenu des analyses (qui est repris dans le programme génératif naissant), mais sur l'épistémologie qu'entraîne l'idée de concevoir une grammaire comme un système constructionnel dédié à la reconstruction logique du savoir sur le langage<sup>5</sup>. On peut brièvement énumérer les points de rupture en les rapportant aux caractères d'un SC tel que les définit Goodman : (a) le choix des éléments primitifs de la grammaire n'est pas réductible à la construction de procédures de découverte des éléments d'une langue (l'analyse distributionnelle) ; (b) une langue (un fragment de langue) peut donner lieu à plusieurs grammaires adéquates et (c) un critère de choix entre plusieurs grammaires est leur caractère « révélateur » : telle grammaire fait apparaître ou met en valeur des aspects insoupçonnés de l'objet alors que telle autre les laisse dans l'ombre de la description. On sait que (c) a toujours été le talon d'Achille du programme génératif, et en particulier des différentes versions que Chomsky va lui donner. C'est aussi sur ce point que Chomsky suit et se sépare de la notion de SC.

### 1.1.2. Le choix d'une grammaire

Pour Goodman, la fonction première d'un SC est une fonction de connaissance : le fait de formaliser une description doit permettre de mettre à jour une organisation interne que la description pré-théorique rate parce qu'elle est imprécise, redondante et embrouillée. « The primary function of a constructional system [...] is to exhibit a network of relationships obtaining in the subject-matter » (Goodman, 1972 : 18). C'est exactement ce type d'argument que Chomsky place en prémisses de sa critique des grammaires syntagmatiques au chapitre 5 de *Syntactic Structures* : « La preuve la plus décisive de l'inadéquation d'une théorie linguistique consiste à montrer qu'elle ne peut absolument pas s'appliquer à une langue naturelle. Une preuve plus faible, mais parfaitement suffisante, de son inadéquation, serait de montrer que la théorie ne s'applique qu'avec lourdeur, c'est-à-dire que toute grammaire construite selon cette théorie est fort complexe, forgée *ad hoc*, et non révélatrice » (Chomsky [1957], 1969 : 39). L'introduction de la notion de transformation dans le même chapitre est faite au nom de cette argumentation. Chomsky prend l'exemple du passif ; selon lui, une grammaire syntagmatique est incapable de faire ressortir la relation qui unit la phrase active et la phrase passive, alors qu'une transformation exprime directement cette relation<sup>6</sup>.

Chomsky propose également un autre type de critère et c'est là qu'il s'écarte de Goodman et de la logique des systèmes constructionnels. C'est un critère externe d'adéquation qu'il emprunte au programme cognitif qu'il identifie à la construction des grammaires : une grammaire est d'autant meilleure qu'elle incarne une hypothèse réaliste sur la faculté de langage qui est appréhendée elle-même comme une grammaire<sup>7</sup>. C'est ce qui explique le singulier « a correct generative grammar » dans le fragment que nous citons en (1). Milner donne une analyse particulièrement claire des deux types de critères et de leur logique propre : l'introduction du critère cognitif devrait permettre « de choisir entre deux théories T1 et T2 en faisant usage de cette instance indépendante, et sans avoir à se limiter à des critères intrinsèques -- élégance, simplicité, etc. » (Milner, 1981 : 303). C'est précisément en faisant appel aux deux critères d'expressivité et de plausibilité psychologique<sup>8</sup> censés caractériser une bonne grammaire que les tenants des grammaires syntagmatiques attaqueront la forme transformationnelle des grammaires et montreront la supériorité de la forme syntagmatique.

---

<sup>5</sup> Sur la continuité de contenu entre le programme génératif et le structuralisme, voir par exemple (Encrevé, 1997). Sur la stratégie socio-institutionnelle de rupture menée par Chomsky et son groupe, voir (Murray, 1993).

<sup>6</sup> L'analyse du passif comme une relation entre deux phrases sera rapidement abandonnée (entre autres Bresnan, 1982, chap. 1 : *The Passive in Lexical Theory*).

<sup>7</sup> Goodman commettra plusieurs articles particulièrement ironiques critiquant cette hypothèse. « I am not denying that the case of "eager" and "easy" may have important features not shared by the case of 'rat' and 'cat'. I am asking for the grounds for the inference from such features, or from other peculiarities of words, to innate ideas » (1972 [The emperor's new ideas] : 77).

<sup>8</sup> Plus précisément, à la capacité d'une grammaire à fonder « des modèles psychologiquement réalistes de l'acquisition, de la compréhension et de la production du langage » (Bresnan et Kaplan, 1982 : xviii).

Si la notion de SC donne la forme générale et les principes de construction de la notion de grammaire, il reste à les mettre en œuvre. La notion de générativité et de récursivité sont ici décisives : « si une grammaire comporte des mécanismes récursifs, quels qu'ils soient, elle produira des phrases en nombre infini » (Chomsky [1957], 1969 : 27). Grâce à la récursivité, une grammaire se présente comme un système permettant d'engendrer la description structurale d'un nombre infini de phrases<sup>9</sup>.

## 1.2. Les grammaires formelles

Le programme de construction des grammaires est inséparable de la recherche du meilleur système pour décrire les langues (« a correct generative grammar »). Dans l'horizon des SC, Chomsky introduit la formalisation des grammaires comme un des instruments de sélection ; il pose clairement que la formalisation permet de réfuter ou de corroborer une forme de grammaire ou une analyse particulière. Ce recours à la formalisation va constituer un des traits les plus stables du programme génératif (cf. (9) ci-dessous). Chomsky appelle la partie des mathématiques qui permet d'étudier les grammaires « linguistique algébrique ». De nombreux travaux ne tarderont pas à paraître<sup>10</sup>. Un des résultats essentiels se donne sous forme d'une hiérarchie des grammaires. Il est remarquable que Chomsky utilise cette recherche pour critiquer un certain nombre de formes de grammaire, et en particulier la forme syntagmatique, et non pour formaliser la forme qu'il propose et défend : la grammaire transformationnelle.

### 1.2.1. La hiérarchie de grammaires

L'objet de la linguistique algébrique est de caractériser les propriétés des grammaires formelles ou des grammaires syntagmatiques (*Phrase Structure Grammars, PSG*). Une grammaire formelle permet d'engendrer de manière récursive, à l'aide d'un nombre fini de règles (de réécriture), un langage constitué d'un nombre infini d'énoncés possibles.

Plus précisément, une grammaire formelle met en jeu un vocabulaire  $V$  (qui peut être constitué d'un vocabulaire terminal  $V_T$  et d'un vocabulaire non terminal  $V_N$ ). Les suites finies d'éléments du vocabulaire, ou mots, sont réécrites d'après les règles de la grammaire. De la sorte, en partant d'un symbole initial, ou d'un mot initial (qui est un axiome), on obtient, par dérivation, des suites terminales. L'ensemble des suites terminales qui peuvent être produites par une grammaire donnée constitue le langage engendré par la grammaire. Par exemple, les deux règles  $S \rightarrow a S$  et  $S \rightarrow b$  permettent d'engendrer toutes les suites de la forme  $a a \dots a b$  (où le nombre de  $a$  est quelconque).

En apportant des restrictions à la forme des règles, on restreint la classe des langages qui peuvent être engendrés. C'est ainsi qu'a été établie une hiérarchie entre classes de langages ou types de grammaires, connue sous le nom de *hiérarchie de Chomsky*. La formulation la plus communément admise considère quatre types de grammaires. Le type 0 correspond à la classe des langages la plus large, puisque les règles ne connaissent aucune restriction. Les grammaires sensibles au contexte (*Context-Sensitive Grammars, CSG*) forment le type 1 : un symbole unique est réécrit, mais cette réécriture peut dépendre d'un contexte gauche et/ou droit. Une telle dépendance au contexte n'est pas autorisée pour les grammaires indépendantes du contexte (*Context-Free Grammars, CFG*), qui constituent le type 2. Enfin, le type 3 correspond aux langages qui peuvent être reconnus par des automates à nombre fini d'états<sup>11</sup>.

<sup>9</sup> Le terme de description structurale apparaît dans Carnap (*Der Aufbau* [...], section 12) : *Strukturbeschreibung*. Chomsky donne (Post, 1944) comme le prototype de l'idée de théorie générative (voir par exemple Chomsky, 1966 : 83).

<sup>10</sup> Chomsky et Miller (1958), Chomsky (1959), Chomsky (1963), Chomsky et Schützenberger (1963).

<sup>11</sup> A tout mot obtenu par dérivation dans une grammaire, hormis pour les grammaires de type 0, et pour les grammaires qui admettent des règles où un symbole peut être réécrit par le mot vide, on associe un arbre : l'arbre de dérivation. Cet arbre est le vecteur de la description structurale de l'énoncé.

### 1.2.2. Le paradoxe de Chomsky

Chomsky se sert de ce résultat pour rejeter un certain nombre de modèles possibles de grammaire. Il rejette les grammaires de type 3 et les grammaires syntagmatiques indépendantes du contexte. C'est ce second rejet qui est le plus important. En effet, Chomsky pose que la grammaire présumée dans l'analyse en constituants immédiats est une grammaire syntagmatique<sup>12</sup>. L'argument principal fondant le rejet est au départ son manque d'expressivité (cf. plus haut) : Chomsky veut que la grammaire puisse exprimer les propriétés communes à plusieurs types d'entités syntaxiques (par exemple, les propriétés communes à la phrase active et à la phrase passive). Progressivement, une argumentation plus directement empirique se met en place : elle consiste à identifier des falsificateurs possibles des grammaires syntagmatiques et, de façon plus restrictive, des grammaires syntagmatiques CF. Deux ensembles de falsificateurs ont été rassemblés. Le premier est composé de structures générales<sup>13</sup>:

- (3) a) la constituance non bornée (*unbounded branching*),
- b) les syntagmes discontinus,
- c) la discordance entre ordre linéaire et constituance.

Le second a été rassemblé par (Pullum et Gazdar, 1981). Il est composé d'une liste assez hétéroclite de tours particuliers :

- (4) a) l'analyse des comparatives en anglais : *\*that one is wider than this one is wide*,
- b) l'analyse de l'expression en langue naturelle des décimales du nombre  $\pi$ ,
- c) l'analyse de *respectively* (*respectivement*) : *John, Mary, David, ... are a widower, a widow, a widower, ... respectively*,
- d) la structure emboîtée du GV en hollandais :  $NP_1 NP_2 \dots NP_n V_1 V_2 \dots V_n$ ,
- e) l'incorporation en mohawk.

La réforme syntagmatique se construira sur la réduction de ce double ensemble de contre-exemples. Le point qui nous importe ici pour l'histoire du programme génératif est le suivant : la formalisation est investie dans la réfutation des formes possibles de grammaire et non dans la démarche positive de construction de la forme que Chomsky promeut et qu'il va développer : les grammaires transformationnelles. Une grammaire transformationnelle est une forme hybride à plusieurs composantes<sup>14</sup> où l'essentiel des interconnexions est exprimé dans le langage des transformations. Or, les transformations ne donnent pas lieu à une formalisation, mais à une simple définition exprimée en termes connotant un traitement informatique : une transformation est définie comme une procédure sur un input pour donner un output. Avec les transformations, un pan du programme génératif abandonne l'exigence d'explicitation formelle pour se contenter de ce que Milner (1989) appellera l'épistémologie du dispositif<sup>15</sup>, dont on retiendra les trois traits suivants : (a) les règles sont littéralisées et non formalisées, (b) elles reçoivent une interprétation causale (les règles sont la cause des propriétés des énoncés) et non sémantique (les règles décrivent les propriétés des énoncés) et (c) elles sont les seuls éléments qui soient suffisamment détaillés pour être l'objet d'une falsification, ce qui a pour résultat de mettre la forme générale de la grammaire et ses

---

<sup>12</sup> L'analyse en constituants immédiats peut être considérée comme un des acquis de la méthode structurale alors dominante. Le programme génératif conservera les catégories syntagmatiques et lexicales qu'elle a permis de dégager.

<sup>13</sup> Cette liste est présentée dans (Manaster-Ramer et Kac, 1990 : 335). Les auteurs ajoutent un quatrième « phénomène » : la classification croisée. De fait, il ne s'agit pas d'un phénomène particulier, le terme renvoie aux ressemblances entre phrases ou syntagmes, ressemblances que la grammaire transformationnelle a analysées comme des relations : le passif, mais aussi les nominalisations (relation entre groupes nominaux et phrases), etc.

<sup>14</sup> Que Chomsky appelle niveaux : niveau syntagmatique, transformationnel et morpho-phonologique.

<sup>15</sup> Contrairement à Milner (1989), nous tenons que l'épistémologie du dispositif ne caractérise pas tout le programme génératif, mais seulement la version qu'en donne Chomsky et le groupe qu'il dirige.

construits propres (tout particulièrement, les notions de transformation, de catégorie vide, etc.) hors discussion. L'entreprise de recherche formelle portant sur le système adéquat pour une grammaire va se poursuivre sans Chomsky, et même le plus souvent contre Chomsky.

### 1.2.3. Les grammaires CF en informatique

Les CFG ont un destin tout contraire en informatique. Très rapidement, elles donnent la forme standard de la description syntaxique des langages de programmation : la forme *BNF* (*Backus-Naur Form*). Dans la forme BNF, un langage de programmation est syntaxiquement décrit par un ensemble de règles de réécriture CF. De plus, l'approfondissement de la caractérisation mathématique des CFG permet d'établir qu'elles donnent lieu à des algorithmes d'une complexité raisonnable. Ce résultat est investi en compilation, c'est-à-dire dans la traduction de programmes écrits dans un langage de programmation évolué en des programmes écrits en langage machine<sup>16</sup>.

Ce même résultat est repris dans le TAL indépendamment du rejet formulé par Chomsky : les algorithmes d'analyse des langues naturelles standards basés sur des CFG sont définis dans les années soixante. On peut citer les algorithmes de Kuno et Oettinger (1962), l'algorithme CYK (Hays, 1962, Younger, 1967, entre autres) ou l'algorithme de Earley (1970). L'existence de ces algorithmes sera au cœur de la contre-argumentation menée par les tenants de la forme syntagmatique. En effet, ils seront pris comme des modèles psychologiquement plausibles de la compréhension des énoncés en langue naturelle.

### 1.3 Les ATN

La conception des ATN (*Augmented Transition Networks*) est un moment essentiel dans l'histoire qui nous occupe (Woods, 1970) : ils vont avoir une position hégémonique en TAL pendant plus d'une décennie<sup>17</sup>. Ils sont à la fois une implémentation des grammaires transformationnelles et un modèle formel alternatif. Ils doivent remédier aux faiblesses formelles et informatiques des grammaires transformationnelles tout en constituant un outil de description puissant : "The theory of transformational grammar proposed by Chomsky is one of the most powerful tools for describing the sentences that are possible in a natural language and the relationships that hold among them, but this theory as it is currently formalized (to the limited extent to which it is formalized) loses the perspicuousness of the context-free grammar" (Woods, 1970 : 599). Ce moment correspond à l'essor du langage de programmation LISP. En LISP, les données et les programmes sont écrits selon la même syntaxe, celle des structures bien parenthésées. Ce qui autorise un ATN à être à la fois une structure de données et un programme.

Plus précisément, un ATN est un réseau d'automates auquel sont associées des conditions et des actions. Le parcours du réseau d'automates s'effectue tandis qu'on progresse dans la reconnaissance d'une phrase, lue de gauche à droite. Les actions permettent d'affecter des valeurs à des registres<sup>18</sup>. Ces valeurs sont des valeurs atomiques ou des structures bien parenthésées ; elles représentent les propriétés de n'importe quelle partie de la phrase analysée ou elles en donnent la description structurale sous forme arborescente. Elles sont susceptibles de varier lors du parcours des automates. Les conditions, qui portent sur les valeurs affectées aux registres, déterminent quels sont les parcours valides.

Les différentes valeurs affectées aux registres à la fin du parcours d'un automate du réseau sont à la base du calcul d'une valeur unique, qui tient lieu de résultat du parcours de l'automate (de la

---

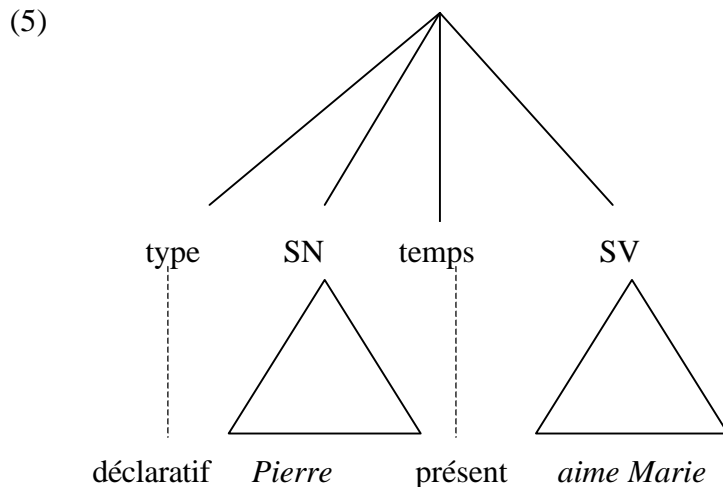
<sup>16</sup> Ainsi, Aho et Ullman reconnaissent la dette de l'informatique envers la linguistique : "The original source for the notion of context-free grammars is Chomsky [1956] and Chomsky [1959]. These grammars were advanced as a way of defining natural languages rather than computer languages" (1977 : 104).

<sup>17</sup> Les auteurs des outils formels et/ou informatiques pour le TAL qui suivront, comme DCG ou FUG, devront se positionner par rapport aux ATN.

<sup>18</sup> De la même manière que les langages de programmation affectent des valeurs aux cases de la mémoire des ordinateurs.



même manière que les sous-programmes calculent un résultat, communiqué au programme qui les appelle). Le résultat final correspond à l'analyse de la phrase lue ; il se présente sous la forme d'un arbre hétérogène. Ce que nous illustrons sous (5)<sup>19</sup> :



Le long succès que les ATN connaîtront dans le domaine du TAL fera oublier leur origine transformationnelle. Ils serviront dans des cadres divers et variés. Un ATN, en effet, est un outil non contraint, qui a la puissance d'une machine de Turing. Il permet d'associer à un énoncé n'importe quel type de structure, ce qui est très pratique dans une perspective d'application industrielle où il s'agit d'obtenir un output à partir d'un énoncé : traduction, interrogation en langue naturelle, etc.

Les ATN seront la cible de deux critiques qui conduiront à leur abandon. La première porte sur leur capacité à soutenir des modélisations de la compétence ou de la performance. Ils ne peuvent servir à modéliser une grammaire si on admet qu'une grammaire est un modèle de la connaissance qu'un locuteur a de sa langue et que cette connaissance peut être mise en application indifféremment dans des procédures de reconnaissance ou de production d'énoncés. Mais, ils ne peuvent pas non plus servir à modéliser les procédures de reconnaissance caractéristiques de la performance si on admet que la grammaire est une ressource neutre et extérieure à la procédure elle-même : en effet, les connaissances grammaticales sont mêlées aux opérations constitutives de la reconnaissance<sup>20</sup>. Ce défaut (radical) est lié à leur nature procédurale, qui est la cible d'une deuxième critique interne au TAL. Le développement des systèmes experts a montré l'intérêt pratique et théorique de distinguer la base de connaissances du moteur d'inférences, et plus généralement les connaissances des procédures. Cela devient un critère de bonne formation des systèmes informatiques : un système est jugé d'autant meilleur que la partie procédurale est plus générale et indépendante de connaissances particulières.

Néanmoins, les ATN enrichissent le répertoire des outils descriptifs maniés par les grammaires : l'affectation de valeurs à des registres préfigure les couples attribut/valeur qui formeront les

---

<sup>19</sup> Le processus d'analyse à l'aide d'un ATN est qualifié par Woods de « reconnaissance transformationnelle ». Cela signifie que le résultat de l'analyse correspond à la structure profonde de la phrase analysée. L'idée selon laquelle plusieurs structures de surface peuvent partager une même structure profonde apparaît comme le contenu essentiel retenu par les ATN des grammaires transformationnelles.

<sup>20</sup> Ce qui fonde cette critique va rester constant dans le programme génératif. On en retrouve les grandes lignes dans l'introduction à HPSG : « grammars that are to fit into realistic models of processing should be completely order-independent. [...] The differences between, say, comprehension and production should be explained by a theory that posits different kinds of processing regimes based on a single linguistic description -- a process-neutral grammar of the language that is consulted by the various processors that function in linguistic activity » (Pollard et Sag, 1994 : 12).

spécifications de traits. Et les arbres hétérogènes (cf. (5) ci-dessus) préfigurent les formalisations à venir de la grammaire où cohabitent des informations qui ne se limitent pas à la seule structure syntaxique (cf. (13) ci-dessous).

## 2. Le tournant déclaratif : les grammaires syntagmatiques enrichies et les grammaires d'unification

### 2.1 Le contexte

Le second moment que nous distinguons est marqué par l'enrichissement des grammaires syntagmatiques visant à accroître leur expressivité, la conception du langage de programmation Prolog et une grande inventivité dans la conception de nouvelles formes de grammaire, tant du côté des linguistes (LFG, TAG, GPSG,...) que du côté des informaticiens engagés dans le TAL (DCG, PATR II, FUG,...).

Le développement de l'expressivité des PSG est entrepris très tôt en réponse à l'argumentation de Chomsky en faveur de la forme transformationnelle (Yngve, 1960 et Harman, 1963) ; Harman démontre avec force que Chomsky récuse une version particulière des grammaires syntagmatiques (des grammaires où seule est introduite l'appartenance catégorielle sous la forme de catégorie monadique étiquetant les nœuds), et non les grammaires syntagmatiques en général. Mais, c'est le groupe mené par Gazdar qui transforme l'essai avec GPSG (*Generalized Phrase Structure Grammar*, Gazdar (1982), Gazdar, Klein, Pullum et Sag (1985)). Bresnan donne à l'argument de plausibilité psychologique ses lettres de noblesse avec LFG (*Lexical-Functional Grammar*)<sup>21</sup>. Il est repris par Gazdar qui, pour le développer, fait appel aux résultats acquis dans la théorie de la compilation et dans le TAL<sup>22</sup>. Les algorithmes d'analyse qu'autorisent les grammaires CF constituent des modèles de compréhension beaucoup plus plausibles que ceux qu'autorisent les grammaires transformationnelles<sup>23</sup> :

- (6) The sentences of a natural language can be parsed. We do it all the time. Furthermore, we do it very fast [...]. But for « transformational grammars, it is not known that processing time can be any less than a doubly exponential function of sentence length » (Peters 1979). Transformational grammars thus fail to provide even the beginnings of an explanation for one of the most important and most neglected facts about natural languages: parsing is easy and quick. Sentences of a context-free language are provably parsable in a time which is, at worst, proportional to less than the cube of the sentence length.  
(Gazdar, 1982 : 133)

Par ailleurs, GPSG fait sauter le verrou « bloomfieldien » à l'encontre de la sémantique fermement maintenu par Chomsky à l'orée du programme génératif<sup>24</sup>: chaque règle de réécriture est

---

<sup>21</sup> Le rôle de LFG est essentiel à plusieurs titres. Bresnan et le groupe qu'elle dirige montrent la pertinence d'analyses lexicalisées de nombre de phénomènes traités auparavant en termes transformationnels. Kaplan joue le rôle de passeur de solutions élaborées dans le cadre des ATN vers la linguistique (voir Kaplan, 1995).

<sup>22</sup> « Compiler design for CFL's is a fairly well explored problem (see Aho and Ullman 1972, 1973), but designing compilers for non-CFL's can be grossly more difficult. Nor is this a concern that can be relegated to the field of computer programming; for those who take seriously the thesis of Fodor (1975), language acquisition for a human learner is nothing more or less than the construction of a program to compile the natural language into the human machine code (or whatever intermediate code is used for thinking) » (Pullum et Gazdar, 1982 : 139).

<sup>23</sup> Milner (1981) a raison de souligner que l'hypothèse cognitive n'a pas fourni de critère permettant de choisir entre deux analyses possibles d'un même ensemble de faits linguistiques, mais on voit ici qu'elle joue un rôle moteur pour choisir entre deux formes de grammaires.

<sup>24</sup> « The term "grammar" is understood here in a narrow sense, as pure theory of form, thus incorporating syntax and phonology » (Chomsky, 1975 : 6). Rappelons que la notion de forme logique (FL) est définie dans une grammaire transformationnelle comme le niveau syntaxique susceptible d'être interprété sémantiquement : la FL ne sort pas d'une

accompagnée d'une règle d'interprétation sémantique. Cette implémentation directe du principe général de compositionnalité<sup>25</sup> constitue un argument d'une nature nouvelle en faveur des PSG.

Parallèlement, les recherches en TAL s'affranchissent des ATN. Un moteur puissant d'invention est un principe méthodologique qui a mûri dans la conception des systèmes experts : la déclarativité. C'est au nom de ce principe méthodologique, et indépendamment de toute considération d'adéquation descriptive ou de plausibilité psychologique, que Pereira et Warren, par exemple, défendent un nouveau formalisme grammatical, les DCG (*Definite Clause Grammars*) :

- (7) The greater clarity and modularity of DCGs is a vital aid in the actual development of systems of the size and complexity necessary for real natural language analysis. Because the DCG consists of small independent rules with a declarative reading, it is much easier to extend the system with new linguistic constructions, or to modify the kind of structures which are built. (Pereira et Warren, 1980 : 270)

Les DCG sont historiquement indissociables du langage de programmation Prolog, qui lui-même naît de l'objectif d'effectuer l'analyse syntaxique des langues naturelles et d'une traduction des grammaires CF dans le calcul des prédicats du premier ordre. De la même manière que la définition des grammaires coïncide avec la vogue du langage Algol, que les ATN sont plongés dans LISP, les formalismes déclaratifs accompagnent le succès de Prolog. Ce remarquable parallélisme entre formalismes grammaticaux et langages de programmation se confirmera dans le troisième moment où l'on voit les langages orientés objets accompagner les grammaires de construction.

La tentative de Shieber (1986) marque un tournant dans l'histoire du contact entre linguistique et informatique : il s'agit ni plus ni moins que de dégager les traits formels communs aux différents modèles grammaticaux non transformationnels et aux formalismes déclaratifs. C'est ce corps commun que Shieber dénomme (en se conformant à la grammaire de la dénomination des systèmes grammaticaux, cf. *grammaire transformationnelle*) *grammaire basée sur l'unification* ou, de façon abrégée, *grammaire d'unification*. Si linguistique et informatique (plus particulièrement TAL) sont des programmes distincts menés par des objectifs indépendants (ce que reconnaît Shieber dès l'abord), il n'en reste pas moins que se dessine une convergence qui constitue un des traits du troisième moment que nous distinguerons dans l'histoire du contact entre linguistique et informatique.

## **2.2 La réforme : l'enrichissement des grammaires syntagmatiques**

Les recherches qui s'incarnent dans différentes grammaires syntagmatiques partagent un même refus de la forme transformationnelle de la grammaire, qui est analysée comme un dévoiement des objectifs et des idéaux du programme génératif. Ce refus s'appuie sur quatre lignes d'argumentation. La première est empirique : la réduction des contre-exemples aux grammaires syntagmatiques. Pullum et Gazdar (1981) montrent que les cinq tours (4) ont été incorrectement analysés ; ils ne peuvent pas être tenus pour des contre-exemples à la thèse selon laquelle les langues naturelles sont CF<sup>26</sup>. Les trois structures (3) sont plus « coriaces » : elles font l'objet de multiples tentatives de réduction que nous évoquerons plus bas. La seconde porte sur l'expressivité des grammaires syntagmatiques : les concepteurs de grammaires montrent qu'on peut les enrichir de telle sorte qu'elles permettent l'expression claire et distincte des principes organisateurs des langues. La troisième est formelle : les grammaires syntagmatiques enrichies sont définissables formellement à la différence des grammaires transformationnelles<sup>27</sup>. La quatrième porte sur la plausibilité

---

« théorie pure de la forme ».

<sup>25</sup> Repris au programme montagovien via l'hypothèse connue sous le nom de « rule to rule hypothesis » proposée par Bach (1976).

<sup>26</sup> Cet article est un morceau d'anthologie d'humour.

<sup>27</sup> « The mathematical properties of the resulting baroque systems [les grammaires transformationnelles] are almost

psychologique de la forme grammaticale : on a vu ci-dessus les arguments avancés par Gazdar. La proposition de Chomsky selon laquelle « the theory of phrase structure does not suffice for the characterization of linguistic competence of what the language user knows. It is suggested that linguistic theory requires a new and more abstract level of description, the level of grammatical transformations » (Chomsky, 1975 : 8) peut donc être tenue pour invalidée. Les tenants des grammaires syntagmatiques peuvent se présenter comme les continuateurs les plus fidèles du programme génératif.

Enfin, elles partagent un style épistémologique qui privilégie l'invention formelle prudente plutôt que la rupture ; c'est ce qu'exprime emblématiquement ce passage de Gazdar :

- (8) The obvious thing to do if natural languages were ever shown not to be CFL's in the general case would be to start exploring minimal enhancements of expressive power to determine exactly what natural languages call for in this regard and how it could be effectively but parsimoniously provided in a way that closely modelled human linguistic capacities. (Pullum et Gazdar, 1981 : 171 ; nous soulignons).

Nous décrivons brièvement dans ce qui suit certains des outils formels qui ont permis d'augmenter le pouvoir expressif des PSG.

### 2.2.1. Les catégories comme ensemble de traits : GPSG

L'innovation clef est la définition des catégories syntaxiques comme des ensembles de spécifications de traits, où les traits sont définis comme des couples <attribut, valeur>. L'innovation se nourrit de l'emploi informel des traits développé dans le cadre des grammaires transformationnelles. Il s'agit essentiellement de la décomposition des catégories majeures en deux traits binaires N et V, ainsi que de la décomposition des catégories syntagmatiques dans le cadre de la théorie X-barre en un type catégoriel et un niveau syntagmatique. Ainsi, la catégorie syntagmatique GN est analysée et représentée par {<N,+>,<V,->,<BARRE,2>}, alors que la catégorie lexicale N est analysée et représentée par : {<N,+>,<V,->,<BARRE,0>}. Toutes les propriétés des expressions peuvent être exprimées sous forme de traits : un GN singulier à la troisième personne et à l'accusatif est analysé et représenté par {<N,+>,<V,->, <BARRE,2>, <PER,3>, <PLU,->,< CAS,ACC>}<sup>28</sup>.

Cette définition des catégories autorise l'écriture de règles (qui, en GPSG, sont des règles de dominance immédiate ou des règles de précédence linéaire<sup>29</sup>) à plusieurs niveaux de généralité selon le degré de sous-spécification des catégories. Par exemple, la règle <SOUS-CAT> < ¬ <SOUS-CAT> est une règle de précédence linéaire qui permet d'ordonner toute expression sous-catégorisée après l'expression qui la sous-catégorise. L'introduction de catégories sous-spécifiées ouvre la voie à une description hiérarchique des objets, qui sera développée dans HPSG, la relation de subsomption entre catégories définissant la hiérarchie. En effet, une règle sous-spécifiée rassemble une famille de règles plus spécifiées ou complètement spécifiées.

---

entirely unknown: we are ignorant, for example, as to whether ungrammaticality with respect to such grammars is decidable, i.e. given an arbitrary string on the terminal vocabulary, no way is known of proving that that string is not generated by the grammar. In this situation, claims by grammarians to the effect that such and such a string of words cannot be generated by their grammar merely reflect their intuitions about the apparatus they are using » (Gazdar, 1982 : 131).

<sup>28</sup> A bien des égards, l'innovation consiste à généraliser l'emploi des symboles complexes comme étiquettes de tous les nœuds de l'arbre syntagmatiques et non de les cantonner aux nœuds pré-terminaux comme dans la composante syntagmatique des grammaires transformationnelles proposées dans *Aspects* (Chomsky, 1965). « Harman (1963) deserves the credit for first seeing the potential of PSGs incorporating complex symbols » (Gazdar, 1982 : 134).

<sup>29</sup> D'où le nom de format DI/PL.

Par ailleurs, GPSG introduit des outils supplémentaires, en particulier les principes<sup>30</sup>. Trois principes rendent compte de l'essentiel des phénomènes liés à la structuration syntaxique des langues naturelles : la Convention des traits de tête rend compte de l'endocentricité des syntagmes (le syntagme partage les valeurs de certains traits, définis une fois pour toutes, avec sa tête), le Principe du contrôle et de l'accord traite de façon unifiée des phénomènes d'accord et le principe des traits de pied permet le traitement des dépendances à distance, ce qui préserve le résultat essentiel acquis dans le cadre des grammaires transformationnelles, le traitement local de la dépendance. Les deux derniers principes requièrent une généralisation du formalisme des traits, puisque la valeur d'un trait peut être une catégorie.

GPSG initie un déplacement, que va systématiser HPSG : on passe d'une approche où la dérivation est le vecteur des généralisations linguistiques à un système où sont déclarés des principes de bonne formation des différentes entités permettant d'analyser les énoncés<sup>31</sup>. C'est en ce sens que la recherche dont GPSG est le vecteur converge avec les recherches menées dans le TAL et la représentation des connaissances : la théorie syntaxique se donne explicitement dans une forme grammaticale que l'on peut qualifier de déclarative.

GPSG ne clôt pas le problème posé par les structures qui ont été identifiées comme les falsificateurs possibles des grammaires syntagmatiques CF. La discordance entre ordre et constituance est portée au rang de principe structural dans le format de la grammaire en distinguant règle d'ordre et règle de dominance. La constituance non bornée est traitée par l'introduction d'un type particulier de règles, qui fait usage de l'étoile de Kleene<sup>32</sup> :  $A \rightarrow B C^*$ . Cette notation signifie en fait que l'on a un nombre infini de règles :  $A \rightarrow B$ ,  $A \rightarrow B C$ ,  $A \rightarrow B C C$ , ... Il en résulte que l'usage de l'étoile de Kleene dans les parties droites des règles (de même que l'introduction des métarègles) représente plus qu'un « minimal enhancement » des PSG : elles rompent avec une caractéristique essentielle des PSG, la génération de langages infinis par des grammaires finies. Cette rupture n'est pas indispensable, comme le montreront entre autres les grammaires d'arbres polychromes (cf. ci-dessous). Les constituants discontinus ne trouvent pas de solution naturelle dans le cadre de GPSG, malgré les tentatives entre autres de Pullum (1982) et Ojeda (1987).

### 2.2.2 La redéfinition de la composition des arbres : grammaires d'arbres adjoints et grammaires d'arbres polychromes

L'enrichissement ne concerne pas seulement la forme des catégories. Deux cadres grammaticaux redéfinissent l'opération de composition des arbres (à laquelle GPSG ne touche pas). Il s'agit de deux grammaires où la notion d'arbre est centrale : les grammaires d'arbres adjoints (*Tree Adjoining Grammars*, TAG, Joshi, 1985) et les grammaires d'arbres polychromes (GAP, Cori et Marandin, 1993). Les TAG reposent sur une séparation des grammaires en deux sortes d'arbres primitifs (les arbres initiaux et les arbres auxiliaires) et une opération de composition, l'adjonction, qui vient insérer des arbres auxiliaires à l'intérieur des arbres syntaxiques (alors que, dans les grammaires traditionnelles, l'opération de substitution place des arbres primitifs aux feuilles des arbres syntaxiques). Cela permet un traitement non transformationnel des relations à longue distance.

Une GAP est composée d'un ensemble d'arbres dont les branches sont ordonnées par rapport à un pivot (qui peut être la tête dans un constituant endocentrique). La composition d'arbres a les deux propriétés suivantes : (a) elle conserve l'ordre des branches par rapport au pivot et (b) elle écrase les niveaux hiérarchiques lorsqu'elle met en jeu des catégories de pivot identique. Grâce à ces deux propriétés, on peut engendrer des constituants à branches multiples sans recours à l'étoile de

---

<sup>30</sup> A côté des restrictions de cooccurrence de traits, des spécifications de traits par défaut et des métarègles.

<sup>31</sup> C'est ce que Milner appelle « lois à l'indicatif ». « Cet indicatif ne saurait être confondu avec le constat de ce qui est observable dans la réalité matérielle : la démarcation dans la langue ne répond pas toujours à la réalité des formes préférées. [...] il n'y a pas de règles du langage, mais des lois et des propriétés » (1989 : 253).

<sup>32</sup> C'est par exemple le cas dans les règles de la coordination.

Kleene<sup>33</sup> et obtenir les effets d'intrication caractéristiques des tours incidents<sup>34</sup>. Il est intéressant de noter que les GAP, comme les grammaires DI/PL mais d'une manière différente, ne présupposent pas que les relations d'ordre entre constituants soient réductibles aux relations d'appartenance syntagmatique.

De fait, on touche là à un problème fondamental de la théorie syntaxique puisqu'il met en jeu la notion même de syntagme telle qu'elle a été façonnée dans l'analyse en constituants immédiats, conservée par le programme génératif et portée au rang de principe organisateur a priori dans les différents modèles transformationnels. La notion de syntagme comme un mixte indissociable de relations de dépendance (par exemple, entre une tête et les constituants qu'elle sous-catégorise), relation d'adjacence sur la chaîne et un domaine d'opacité perd de son évidence. Dès lors qu'on éclate ces différentes facettes du syntagme, le cadre formel fourni par les grammaires syntagmatiques se défait. Et c'est bien ce que l'on observe, par exemple, dans la théorie syntaxique sous-jacente à HPSG : la notion de syntagme est dissoute dans celle de signe, les problèmes d'ordre sont disjoints de ceux qui relèvent de la mise en relation entre une tête et ses dépendants et les relations syntagmatiques (par exemple la relation de c-commande) voient leur portée explicative très fortement relativisée.

Il est donc remarquable, mais non surprenant, que la mise en cause de la notion fondamentale sur laquelle reposent les grammaires syntagmatiques provienne des grammaires qui en ont exploré le plus sérieusement les implications formelles et empiriques.

### 2.3. Les grammaires d'unification

Nous présentons ci-dessous quelques-uns des aspects des formalismes déclaratifs qui caractérisent ce second moment. Sous les ressemblances que maximise Shieber apparaît une différence de fond : elle oppose les formalismes qui reposent sur une idée de grammaire (DCG, PATR) et un cadre comme FUG (qui préfigure HPSG), où la forme organisatrice n'est plus la grammaire, mais un dispositif non spécifique de représentation des connaissances<sup>35</sup>. Linguistiquement, cela implique que la syntaxe n'est plus qu'une dimension caractéristique parmi d'autres des expressions langagières.

#### 2.3.1 Prolog et les DCG

C'est dans l'objectif d'effectuer l'analyse syntaxique des langues naturelles que des chercheurs, au premier rang desquels Colmerauer, ont défini le langage Prolog (Colmerauer et al., 1973)<sup>36</sup>. Il s'agissait d'exploiter la similitude entre la reconnaissance d'un énoncé et la démonstration d'un théorème. Si les règles de la grammaire et les unités du lexique sont vues comme des axiomes, les théorèmes sont les énoncés qui font partie de la langue. Pour mettre en œuvre ce rapprochement, on représente les règles de la grammaire et les unités du lexique par des formules logiques.

Les formules que traite le langage Prolog sont des clauses de Horn, ou clauses définies (*Definite Clause*). De fait, on ne parle de DCG que lorsque les grammaires logiques sont augmentées. Pour les promoteurs des DCG (Pereira et Warren, 1980), en effet, il fallait faire au moins aussi bien que

<sup>33</sup> Constituants coordonnés, mais aussi constituants adjoints : cf. *le ski dans les Alpes à trois mille mètres en été*.

<sup>34</sup> Par exemple : *Pierre, dit-il, est venu hier* (Cori et Marandin, 1995). L'intérêt formel est également que l'on n'a pas à concevoir des structures discontinues pour traiter des constituants incidents.

<sup>35</sup> Shieber gomme cette différence, comme il gomme les différences qui existent entre les grammaires syntagmatiques enrichies et les formalismes déclaratifs. Il y a une part de politique scientifique dans cette entreprise. Pour la compréhension des enjeux actuels, il est préférable de marquer ces différences. Par exemple, l'unité de base d'une grammaire est une règle pour GPSG, DCG et PATR, alors que c'est une structure de traits pour FUG ; la forme de la représentation d'une phrase est un arbre pour GPSG, alors que c'est une structure de traits pour DCG, PATR et FUG. Par exemple encore, l'unification joue un rôle marginal dans GPSG, alors qu'elle joue un rôle central dans DCG, PATR et FUG.

<sup>36</sup> Prolog a servi à d'autres applications, essentiellement liées à l'intelligence artificielle, mais aussi aux bases de données.

les ATN, tout en étant déclaratif<sup>37</sup>. Ils ajoutent donc des traits aux règles, ce qui permet de véhiculer des valeurs, comme cela s'effectue dans les ATN à l'aide des registres. Ces valeurs sont soit des valeurs atomiques, soit des termes Prolog qui correspondent à des structures arborescentes, dont certaines feuilles peuvent être étiquetées par des variables. Elles peuvent conditionner l'application des règles, et elles donnent la description des expressions (syntagmes ou phrases).

Les démonstrations, en Prolog, s'effectuent selon le mécanisme de l'unification : afin de démontrer un but (par exemple qu'une suite de mots constitue une phrase), on cherche à l'unifier avec un fait (ici une information lexicale) ou avec le conséquent d'une implication (ici la partie gauche d'une règle). C'est de là que viendra la notion de grammaire d'unification, bien que le terme ne soit pas encore forgé.

### 2.3.2 Les structures de traits

De nouveaux formalismes ont suivi les DCG, dont le prototype est PATR II (*Parse and Translate*, Shieber). Il s'agissait de maintenir l'idéal déclaratif des DCG, tout en apportant des améliorations. Les améliorations consistent essentiellement en une prise de distance par rapport à Prolog et à la logique des prédicats du premier ordre. En effet, il y a une certaine rigidité dans la syntaxe de Prolog, qui oblige notamment à donner une valeur (éventuellement indéterminée) à tous les attributs, et à maintenir un ordre strict entre les attributs ou leurs valeurs dans les termes.

Les structures de traits diffèrent des termes, auxquels elles se substituent, essentiellement par le fait qu'elles représentent de l'information partielle sur un objet. Ainsi, deux structures de traits différentes peuvent décrire un même objet. L'unification prend alors un nouveau sens : elle devient une opération qui vérifie si deux structures de traits sont compatibles, autrement dit si elles sont susceptibles de décrire le même objet et, dans le cas positif, construit la structure minimale qui regroupe les informations contenues dans les deux structures de départ.

La transmission des valeurs de trait s'effectue dans les ATN selon un mode procédural. Par exemple, le nombre (singulier ou pluriel) d'un verbe devient le nombre du syntagme verbal, puis éventuellement de la phrase par affectations successives d'une valeur aux registres de nombre associés au GV et à la phrase. Dans les formalismes déclaratifs, on parle plutôt de partage des valeurs que de transmission. Les DCG ont recours à des variables logiques. Les structures de traits font appel à d'autres outils, qu'il est plus délicat de formaliser<sup>38</sup>. Deux versions ont été proposées : l'une fait appel à des équations qui contraignent les règles de réécriture, l'autre à des graphes qui sont en général des graphes sans circuit : le partage des valeurs, encore appelé *réentrance*, est marqué par le fait que deux chemins dans un graphe peuvent converger vers un même sommet.

Il y a une grande ressemblance de famille entre DCG et PATR II. L'un et l'autre s'appuient sur une grammaire indépendante du contexte : la grammaire CF définit un ensemble de squelettes qui sont enrichis par les termes, les contraintes équationnelles ou les structures de traits. Dans les deux cas, la structure Prolog ou la structure de traits sert à exprimer la description structurale et non la grammaire elle-même.

### 2.3.3 La dissolution des grammaires dans les structures de traits

Les grammaires d'unification fonctionnelle (*Functional Unification Grammars*, FUG) de Kay (1985) constituent un tournant, car elles effacent la distinction entre le langage de la description et les objets décrits. Les structures de traits prennent le pas sur les grammaires et leurs règles. Les règles sont incluses dans les structures de traits, et une grammaire n'est elle-même qu'une structure de traits obtenue par la disjonction de structures plus élémentaires, qui chacune correspond à une ou plusieurs règles. La disjonction devenant une opération autorisée sur les structures de traits, cela complexifie la nature de celles-ci, d'autant plus que d'autres augmentations sont apportées. Il en

---

<sup>37</sup> Une même DCG peut servir à la reconnaissance ou à la génération.

<sup>38</sup> Les descriptions linguistiques font souvent usage d'une version non formalisée des structures de traits, notées par des matrices dans lesquelles le partage des valeurs est marqué par des "boîtes" numérotées.

résulte l'impossibilité de représenter simplement les structures par des graphes et la nécessité d'étendre l'opération d'unification, ce qui donne lieu à des tentatives formelles variées (Kasper et Rounds, 1986, entre autres).

L'importance historique de FUG ne tient nullement à ses options en théorie linguistique (le fonctionnalisme), de même que les ATN ont eu une importance qui va bien au-delà de leur inspiration transformationnelle. Ce qui compte, c'est que les structures de traits sont mises à contribution pour décrire les différentes dimensions des expressions langagières et que ce sont les mêmes outils formels qui sont utilisés quelle que soit la dimension.

De ce point de vue, il est clair que le formalisme adopté doit être le plus puissant possible. Les formalismes que l'on a proposés pour représenter toute espèce de connaissance humaine ont cette puissance maximale. Les critères de choix du formalisme se trouvent donc profondément modifiés. En particulier, le critère qui veut que le formalisme doit être restreint par les propriétés formelles de l'objet à représenter est remplacé dans une grammaire multidimensionnelle par un critère qui a un effet opposé : un formalisme sera d'autant meilleur qu'il est moins contraint et donc plus à même de permettre la modélisation de dimensions formellement hétérogènes.

### 3. Le tournant représentationnel : les grammaires de construction

#### 3.1. Le contexte

L'épistémologie des systèmes constructionnels est fondamentalement une logique de la découverte : la formalisation a d'abord pour fonction de clarifier et de mettre au jour l'insu des savoirs descriptifs. Chomsky l'énonce clairement au seuil du programme génératif ; Pollard et Sag (1994) le reprennent en introduction à l'exposé de l'état standard de HPSG (*Head-driven Phrase Structure Grammar*) :

- (9) Precisely constructed models for linguistic structure can play an important role, both negative and positive, in the process of discovery itself. By pushing a precise but inadequate formulation to an unacceptable conclusion, we can often expose the exact source of this inadequacy and, consequently, gain a deeper understanding of the linguistic data. More positively, a formalized theory may automatically provide solutions for many problems other than those for which it was explicitly designed. Obscure and intuition-bound notions can neither lead to absurd conclusions nor provide new and correct ones, and hence they fail to be useful in two important respects. I think that some of those linguists who have questioned the value of precise and technical development of linguistic theory have failed to recognize the productive potential in the method of rigorously stating a proposed theory and applying it strictly to linguistic material with no attempt to avoid unacceptable conclusions by *ad hoc* adjustments or loose formulation.

(Chomsky, 1957 : 5), cité par (Pollard et Sag, 1994 : 7-8)

Ce credo épistémologique a montré toute son efficacité dans les approches de « la forme » des langues (morphosyntaxe et phonologie) ; on doit porter au crédit du programme que Chomsky a lancé et animé une avancée empirique considérable. De plus, il a eu un impact décisif en dehors du champ arpenté par les grammairiens. Le credo chomskyen a été repris dans d'autres « façons de prendre le langage » (pour reprendre une expression de Goodman) : la sémantique formelle, les théories du discours, la pragmatique, l'usage situé du discours (ethnométhodologie par exemple), la cognition située (théorie et sémantique des situations, par exemple), etc. Les années quatre-vingt voient émerger des analyses qui se trouvent au confluent de ces différentes disciplines. L'analyse des questions par Ginzburg (1995) est à ce titre exemplaire : elle intègre des résultats et des notions tirés de la syntaxe formelle, de la sémantique formelle mais aussi de l'analyse de conversation et de la théorie des situations. Et c'est cette intégration qui constitue un défi posé aussi bien à la théorie de la



grammaire qu'aux dispositifs de traitement du langage. C'est ce que reconnaît Kay, le concepteur de FUG :

- (10) In practice I take it that the factors that govern the production of a sentence typically come from a great variety of different sources, logical, textual, interpersonal and so forth. In general, each of these, taken by itself, underdetermines what comes out. When they jointly overdetermine it, there must be priorities enabling a choice to be made among the demands of the different sources. When they jointly underdetermine the outcome, the theory must provide defaults and unmarked cases. The point is that we must be prepared to take seriously the claim that language in general, and individual utterances in particular, fill many different *functions* and that these all affect the theory, even at the syntactic level.  
(Kay, 1985 : 252)

La notion de grammaire est bouleversée : ses formes possibles et son rapport à la manière de prendre le langage (la compétence). Le programme génératif héritait d'une ontologie unidimensionnelle construite dans le mouvement structuraliste (morphème, constituant, phrase) ; les grammaires que nous avons considérées jusqu'ici en reconstruisent l'ordre sous la notion de *syntaxe*. Le programme génératif des années quatre-vingt dix hérite d'objets multidimensionnels où chaque dimension apparaît irréductible aux autres. On passe d'objets à deux facettes (les propriétés combinatoires et la constituance phonologique) à des objets à plusieurs facettes : les facettes formelles, mais aussi les propriétés combinatoires sémantiques, la valeur pragmatique et la charge discursive. Cette multidimensionnalité des objets linguistiques a reçu le nom de *construction*<sup>39</sup> et les systèmes en charge de les décrire celui de *grammaire de construction* :

- (11) Constructions may specify, not only syntactic, but also lexical, semantic, and pragmatic information; lexical items, being mentionable in syntactic constructions, may be viewed, in many case at least, as constructions themselves; and constructions may be idiomatic in the sense that a large construction may specify a semantics (and/or pragmatics) that is distinct from what might be calculated from the associated semantics of the set of smaller constructions that could be used to build the same morphosyntactic object.  
(Fillmore *et al.*, 1988 : 501)

Si on définit une grammaire comme l'intégration de ces dimensions hétérogènes, la tâche des constructeurs de système est d'articuler ces dimensions. De ce point de vue, les recherches (pratiques et formelles) menées dans le domaine de la représentation des connaissances en intelligence artificielle constituent un répertoire de notions et d'outils précieux. C'est ainsi qu'il faut lire les remerciements de Pollard et Sag (cf. (12) ci-dessous) ; on les comparera à la citation de Chomsky que nous avons placée aux débuts de notre histoire (cf. (2) ci-dessus) : la distinction entre mathématiques et informatique n'y est plus de mise lorsqu'il s'agit de concevoir « the internal architecture of the system that the linguistic types form » :

- (12) We want to emphasize the extent to which HPSG is intellectually indebted to a wide range of recent research traditions in syntax (principally nonderivational approaches ...), semantics (especially situation semantics) and computer science (data theory, knowledge representation, unification-based formalisms).  
(Pollard et Sag, 1994 : 1)

---

<sup>39</sup> Comme toujours le langage nous joue son tour favori de l'homonymie : cette notion de construction, qui a son histoire dans les approches traditionnelles de la grammaire, est complètement distincte du processus de construction en jeu dans les systèmes constructionnels.

### 3.2 La représentation de la connaissance linguistique : les structures de traits typées

L'objectif de base de l'intelligence artificielle est d'élaborer des méthodes de représentation des connaissances maîtrisées par les êtres humains. Les *frames* (Minsky, 1975) reposent sur l'idée selon laquelle on dispose d'un certain nombre de schèmes, qui décrivent des types d'objets ou de situations. Devant un objet nouveau ou une situation nouvelle, on sélectionne l'un de ces schèmes modélisé comme un *frame* : on attribue des valeurs aux *slots* qui le composent ; un *slot* a un nom (*slot-name*) et une valeur (*slot-value*) qui peut être une valeur immédiate, ou bien un pointeur sur un autre *frame*.

La parenté avec les structures de traits est évidente. On peut même penser que, dans les structures de traits complexes, le fait que la valeur d'un attribut peut être une autre structure de traits, provient de la notion informatique de *pointeur* et d'*adressage indirect*, et vient donc en droite ligne du stockage des informations en machine.

Les *frames* sont des objets informatiques procéduraux qui, selon la logique de LISP, mêlent programmes et données : la valeur affectée à un *slot* peut être obtenue par un *demon*, c'est-à-dire par l'appel à n'importe quelle procédure. Dans une logique plus déclarative, les réseaux sémantiques, introduits en 1968 par Quillian (1968), ont donné lieu à l'élaboration de plusieurs systèmes de représentations de connaissances (entre autres Bobrow et Winograd, 1977, Fahlman, 1979, Brachman et Schmolze, 1984). Ces systèmes ont eu des versions de plus en plus formalisées, à base essentiellement de graphes, dans lesquels les arcs marquent des relations entre objets ou concepts représentés par les sommets. Les arcs, comme les sommets, peuvent être de nature diverse.

Certains arcs, étiquetés "IS-A" (*is an instance of*) ou "AKO" (*a kind of*) jouent un rôle particulier. Ils servent à assurer une structuration classificatoire des concepts ou des objets, sous la forme de ce qu'on appellera une hiérarchie d'héritage et qui est en fait un graphe sans circuit. De la sorte, il est possible d'obtenir une représentation des connaissances qui soit économique, parce que spécifiant les propriétés au niveau le plus adéquat. Par exemple, si on sait que *les oiseaux ont des plumes* (niveau adéquat de spécification d'une propriété) et que *les poules sont des oiseaux*, chaque *poule* hérite de la propriété d'*avoir des plumes*.

Les structures de traits typées, dont une des expressions les plus achevées est due à Carpenter (1992), sont une des formalisations possibles de ces systèmes de représentation des connaissances. Ces structures supposent l'existence d'un ensemble de types, structurés selon une hiérarchie d'héritage par la relation d'inclusion. Par exemple, le type *syntagme nominal* est inclus dans le type *syntagme*, qui lui-même peut être inclus dans le type *signe*. On indique, au niveau adéquat de la hiérarchie, quels sont les attributs qui s'appliquent à un type donné (par exemple l'attribut *temps* ne s'applique pas à tous les objets linguistiques), et quel est le type de la valeur associée à un attribut pour un type d'objet donné (on ne peut associer à l'attribut *temps* n'importe quelle valeur). Ces informations, générales, contraignent les structures de traits particulières, car les sommets de celles-ci doivent tous avoir un type<sup>40</sup>.

On obtient un cadre pour exprimer toute espèce de caractérisation du langage ou d'une langue particulière, alors que la seule généralité exprimée par les grammaires CF porte sur la forme (arborescente) des représentations syntaxiques. Ce cadre permet de construire des grammaires multidimensionnelles alors que les grammaires CF ne se prêtent qu'à la représentation formalisée de la forme de ce qui constitue une phrase. Une grammaire CF modélise un ensemble d'objets linguistiques, alors qu'une grammaire exprimée par une structure de traits typée modélise la connaissance que l'on a des objets linguistiques.

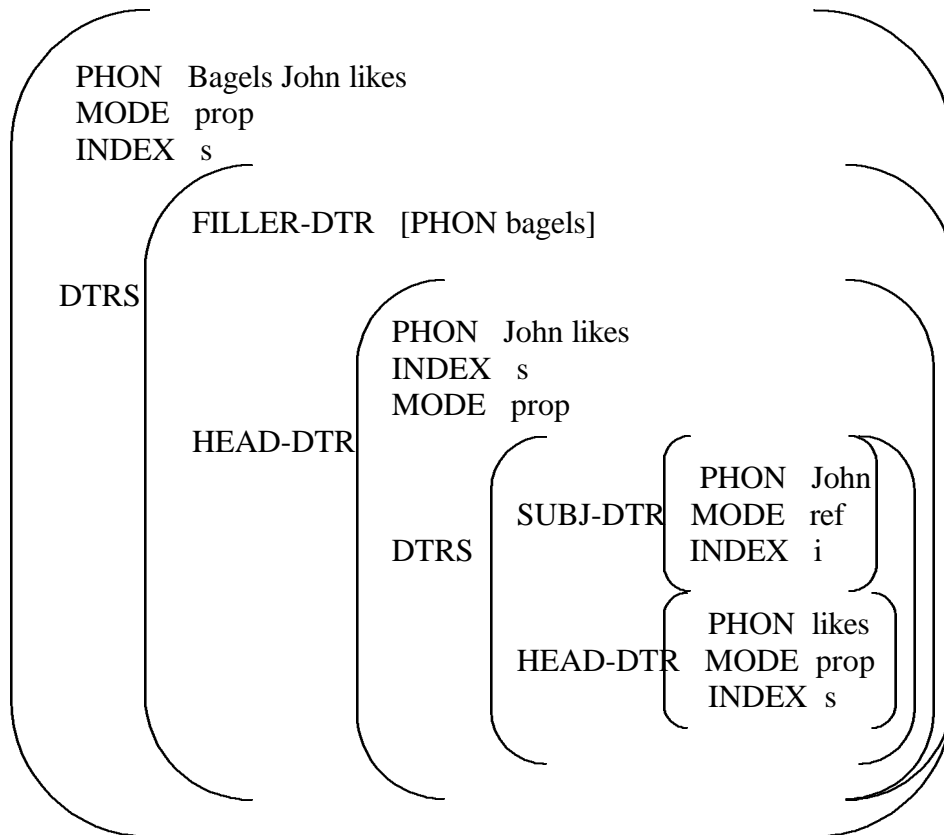
### 3.3. HPSG

HPSG illustre le type de grammaire multidimensionnelle que permettent les structures de traits typées. Elle se présente comme une hiérarchie de signes : « the linguistic types par excellence, the

<sup>40</sup> Les langages orientés objet et, à un niveau plus formel, les types abstraits de données, ont également tiré leur inspiration des *frames* et des réseaux sémantiques et sont largement fondés sur l'héritage des propriétés.

expressions -- or signs (in roughly the Saussurean sense) -- include not only sentences, but also words and subsentential phrases, even multisentence discourses. And a sign is taken to consist not only of a phonetic form, but of other attributes or features as well. That is, we conceive of signs as structured complexes of phonological, syntactic, semantic, and phrase-structural information » (ibid. : 15). Ce système est à la fois caractérisé par l'hétérogénéité des facettes constitutives des objets manipulés et l'uniformité de la modélisation de ces informations. Nous illustrons ce point avec la représentation (13). C'est une représentation partielle, à contenu purement illustratif, qui est un fragment de la grammaire de l'anglais et l'analyse d'une classe de phrases<sup>41</sup>:

(13)



L'hétérogénéité est captée par la classification dans des dimensions distinctes. L'uniformité de la modélisation se manifeste de deux façons. (a) Le modèle de toute unité est construit sur le même patron quelle que soit sa taille : un mot (c'est-à-dire une unité du lexique) est représenté de la même manière qu'un syntagme ou qu'une phrase, voire un discours. Dans (13), le mot *John* présente les mêmes traits que le GN *John* ; le GN *John* présente les mêmes traits que la phrase tête et la phrase enchâssante<sup>42</sup>. (b) Les propriétés de ces entités, quelle que soit leur nature, sont exprimées dans le même format : attribut/valeur. La métaphore de la carte développée par Goodman pour caractériser les systèmes constructionnels résume bien cette tension entre hétérogénéité des informations et homogénéité de la modélisation : « the function of a construction system is not to recreate

<sup>41</sup> L'attribut PHON (phonologie) a pour valeur la réalisation phonologique du signe ; l'attribut MODE (modalité) un type de force illocutoire ; l'attribut INDEX un marqueur de référence (au sens de la DRT). L'attribut DTRS (daughters) a pour valeur la liste des constituants. Dans (13), la structure comprend un constituant détaché (FILLER-DTR) et la tête (HEAD-DTR) qui correspond à une phrase.

<sup>42</sup> « HPSG is "fractal" (structurally uniform as the parts get smaller). Every sign down to the word level (not just the root clause) has features corresponding (inter alia) to phonetic, syntactic, and semantic aspects of linguistic structures » (Pollard, 2000, Lectures on the foundations of HPSG, np.).

experience but rather to map it. [...] A map is schematic, selective, conventional, condensed and uniform » (Goodman, 1972 :15)<sup>43</sup>.

#### 4. Conclusion

Le fragment d'histoire que nous avons reconstruit montre clairement l'importance déterminante des recherches en informatique pour le développement de la linguistique entendue comme entreprise de confection de grammaires qui puissent à la fois intégrer la caractérisation de détail des expressions des langues naturelles, la spécification des universaux formels du langage naturel et le lien aux modélisations possibles de la performance. Les concepteurs de grammaires ont emprunté des notions, des concepts ou des arguments à la théorie de la compilation, aux langages de programmation, à la modélisation des hypothèses du programme cognitiviste et aux dispositifs pratiques de traitement d'information en langue naturelle. Les emprunts ont servi aussi bien à motiver le rejet de certaines formes de grammaires qu'à l'invention de nouvelles formes. D'un côté, Chomsky se sert des premiers résultats de la linguistique algébrique pour rejeter les grammaires syntagmatiques ; les tenants des grammaires syntagmatiques s'appuient sur les résultats de l'informatique pour rejeter la forme transformationnelle. D'un autre, on voit les solutions élaborées dans les ATN, en programmation logique et dans les systèmes de représentations de connaissances être reprises et métamorphosées pour donner lieu à de nouveaux formalismes, dont la caractéristique commune est la déclarativité, et qui évoluent vers une plus grande multidimensionnalité. Les grammaires transformationnelles et les grammaires syntagmatiques soutenaient une même vision du langage dans laquelle l'ordre syntaxique, conçu comme un ordre syntagmatique, est le squelette sur lequel s'ancrent les dimensions non formelles du langage. Les grammaires multidimensionnelles sont le laboratoire où s'élabore une autre vision du langage et de son rapport aux langues.

#### Références

- Aho A.V. et Ullman J.D, 1977, *Principles of Compiler design*, Reading, Massachusetts : Addison Wesley.
- Bach E., 1976, An extension of classical transformational grammar. *Problems in linguistic metatheory, Proceedings of the 1976 Conference at Michigan State University* : 183-224.
- Bobrow D.G. et Winograd T., 1977, An Overview of KRL, A Knowledge Representation Language, *Cognitive Science* 1 : 3-46.
- Brachman R. J. et Schmolze J. G., 1985, An Overview of the KL-ONE Knowledge Representation System, *Cognitive Science* 9 : 171-216.
- Bresnan J., 1982, The Passive in Lexical Theory, [Bresnan J. ed.] *The mental representations of grammatical relations*, Cambridge : MIT Press.
- Bresnan J. et Kaplan R., 1982, Introduction : Grammars as mental representations of language, [Bresnan J. ed.] *The mental representations of grammatical relations*, Cambridge : MIT Press.
- Carpenter B., 1992, *The Logic of Typed Feature Structures*, Cambridge University Press.
- Chomsky N., 1956, Three Models for the Description of Language, *I.R.E. Transactions on Information Theory* 2 : 3, 113-124.
- Chomsky N., 1957, [trad. fr. le Seuil, 1969], *Syntactic structures*, La Haye : Mouton.
- Chomsky N., 1959, On certain formal properties of grammars, *Information and Control* 2 : 2, 137-167.
- Chomsky N., 1963, Formal Properties of Grammars, [D. Luce, E. Bush, E. Galanter, eds] *Handbook of Mathematical Psychology*, John Wiley et Sons.
- Chomsky N., 1965, *Aspects of the Theory of Syntax*, Cambridge : MIT Press.
- Chomsky N., 1966 [trad. fr.], La notion de "règle de grammaire", *Langages* 4 : 81-104.

---

<sup>43</sup> Goodman ajoute « The map not only summarizes, classifies and systematizes, it often discloses facts we could hardly learn immediately from our explorations », ce qui correspond au credo épistémologique rappelé en (9) ci-dessus.

- Chomsky N., 1975, Introduction 1973, *The logical structure of linguistic theory*, Chicago : The University of Chicago Press.
- Chomsky N. et Miller G.A., 1958, Finite State Languages, *Information and Control*, 1, 91-112.
- Chomsky N. et Schutzenberger M.P., 1963, The algebraic theory of context-free languages, [Braffort et Hirschberg, eds] *Computer programming and formal systems, Studies in logic series*, Amsterdam : North-Holland, 119-161.
- Colmerauer A., Kanoui H., Roussel P. et Pasero R., 1973, *Un Système de Communication Homme-Machine en Français*, Groupe de Recherche en Intelligence Artificielle, Université d'Aix-Marseille.
- Cori M. et Marandin J.-M., 1993, Grammaires d'arbres polychromes, *TAL* vol. 34, n° 1 : 101-132.
- Cori M. et Marandin J.-M., 1995, Trees and the Representation of Disjunct Constituents, [G.V. Morrill et R.T. Oehrle, eds] *ESSLLI Conference on Formal Grammar*, Barcelone : 61-74.
- Earley J., 1970, An Efficient Context-Free Parsing Algorithm, *Communications of the Association for Computing Machinery* 13 : 2, 94-102.
- Encrevé P., 1997, L'ancien et le nouveau: quelques remarques sur la phonologie et son histoire, *Langages* n°125 : 100-123.
- Fahlman S.E., 1979, *NETL A system for representing and using real-world knowledge*, Cambridge : MIT Press.
- Fillmore C., Kay P., O'Connor C., 1988, Regularity and idiomaticity in grammatical constructions : the case of *let alone*, *Language* 64 : 501-538.
- Fodor J., 1975, *The language of thought*, New York : Thomas Crowell.
- Gazdar G., 1982, Phrase structure grammar, [Jacobson P. et Pullum G., eds] *The Nature of Syntactic Representation*, 131-186, Dordrecht : D. Reidel Publishing Company.
- Gazdar G., Klein E., Pullum G. et Sag I., 1985, *Generalized Phrase Structure Grammar*, Oxford : Basil Blackwell.
- Ginzburg J., 1995, The semantics of interrogatives, [S. Lappin, ed.] *Handbook of contemporary semantic theory*, Oxford : Basil Blackwell.
- Goodman N., 1951, *The structure of appearance*, Dordrecht : D. Reidel Publishing Co.
- Goodman N., 1972, *Problems and projects*, Indianapolis : Bobbs-Merrill.
- Harman G.H., 1963, Generative Grammars without Transformation Rules: A Defense of Phrase Structure. *Language* 39, 597-616.
- Hays D.G., 1962, Automatic language-data processing, [H. Borko, ed.] *Computer Applications in the Behavioral Sciences*, Englewood Cliffs, New Jersey : Prentice Hall, 394-421.
- Joshi A.K., 1985, Tree adjoining grammars : How much context-sensitivity is required to provide reasonable structural descriptions, [D.R. Dowty, L. Karttunen, A.M. Zwicky, eds] *Natural language parsing*, Cambridge University Press, 206-250.
- Kaplan R.M., 1995, The formal architecture of LFG, [Dalrymple et al, eds] *Formal issues in LFG*, Stanford : CSLI.
- Kasper R. et Rounds W., 1986, A logical semantics for feature structures, *Proceedings of the 24<sup>th</sup> Annual Conference of the Association for Computational Linguistics*, New York : 235-242.
- Kay M., 1985, Parsing in functional unification grammar, [D.R. Dowty, L. Karttunen et A.M. Zwicky eds.] *Natural language parsing*, Cambridge University Press : 251-278.
- Kuno S. et Oettinger A.G., 1962, Multiple-path Syntactic Analyzer, *Information Processing VII*, 1, Amsterdam : North-Holland, 306-311.
- Lakatos I., 1984 [trad. fr.], *Preuves et réfutations*, Paris : Hermann.
- Manaster-Ramer A. et Kac B., 1990, The Concept of Phrase Structure, *Linguistics and Philosophy*, 13, 325-362.
- Milner J.-C., 1981, Linguistique, biologie, psychologie, *Ordres et raisons de langue*, Paris : Le Seuil.
- Milner J.-C., 1989, *Introduction à une science du langage*, Paris : Le Seuil.
- Minsky M., 1975, A framework for representing knowledge, [P. Winston, ed] *The Psychology of Computer Vision*, New York : McGraw-Hill, 211-277.
- Murray S., 1993, *Theory groups and the study of language in North America. A social history*, Amsterdam/Philadelphie : John Benjamins Company.
- Ojeda A., 1987, Discontinuity, Multidominance, and Unbounded Dependency in Generalized Phrase Structure Grammar : some preliminaries, [Huck G. et Ojeda A., eds] *Discontinuous Constituency*, New-York : Academic Press, 257-282.

- Pereira F et Warren D, 1980, Definite Clause Grammars for Language Analysis - A Survey of the Formalism and a Comparison with Augmented Transition Networks, *Artificial Intelligence*, 13 (3) : 231-278.
- Peters S., 1979, How semantics keeps syntax psychologically computable, Paper presented to the Cognitive Studies Seminar, University of Sussex.
- Pollard C. et Sag I. A. , 1994, *Head-Driven Phrase Structure Grammar*, Chicago : University of Chicago Press.
- Post E., 1944, Recursively enumerable sets of positive integers and their decision problems, *Bulletin of the American Mathematical Society* 50 : 284-316.
- Pullum G.K., 1982, Free word order and phrase structure rules, [Pustejovsky J. et Sells P. eds] *Proceedings of the twelfth Annual Meeting of the North Eastern Linguistic Society*, 209-220.
- Pullum G.K. et Gazdar G., 1981, Natural languages and context-free languages, [Savitch W. et al., eds] *The formal complexity of natural language*, Dordrecht : D. Reidel Publishing Company.
- Quillian M.R., 1968, Semantic memory, [M. Minsky, ed.] *Semantic Information Processing*, Cambridge : MIT Press, 216-270.
- Shieber S., 1986, *An introduction to unification-based approaches to grammar*, Stanford : CSLI.
- Woods W.A. , 1970, Transition Network Grammars for Natural Language Analysis, *Communications of the ACM*, 13 : 10, 591-606.
- Yngve V., 1960, A model and an Hypothesis For Language Structure. *Proceedings of the American Philosophical Society*, 104, 444-466.
- Younger D.H., 1967, Recognition and Parsing of Context-Free Languages in Time  $n^3$ . *Information and Control*, 10, 180-208.

**Résumé :**

Les auteurs proposent un essai d'histoire rationnelle d'un aspect du programme génératif dans sa relation à l'informatique, celui qui va de la définition de la notion de grammaire par Chomsky, et en particulier de la caractérisation mathématique des grammaires possibles (la linguistique algébrique), à la conception de grammaires multidimensionnelles qui trouvent dans les systèmes de représentation des connaissances leur fondation formelle. Les auteurs distinguent trois moments : (a) l'émergence du programme génératif et de la notion de grammaire générative ; (b) le mouvement de réforme qui se cristallise dans le refus de la forme transformationnelle de la grammaire et qui voit l'invention de nombreux nouveaux formalismes qui auront tous la caractéristique d'être déclaratifs ; (c) le développement des grammaires multidimensionnelles qui sont chargées d'intégrer les analyses des dimensions formelles (syntaxe et phonologie) et non formelles (sémantique, pragmatique) des langues.