MATHÉMATIQUES ET SCIENCES HUMAINES

AARNE RANTA

Structures grammaticales dans le français mathématique : I

Mathématiques et sciences humaines, tome 138 (1997), p. 5-56

<http://www.numdam.org/item?id=MSH_1997__138__5_0>

Math. Inf. Sci. hum., (35e année, n°138, *1997*, pp.5- 56)

STRUCTURES GRAMMATICALES DANS LE FRANÇAIS MATHEMATIQUE : I.

Aarne RANTA[1]

RÉSUMÉ — *Un système de règles grammaticales est présenté pour analyser un fragment du français permettant l'expression de théorèmes et de preuves mathématiques. Pour cet objectif, on développe une version de la grammaire de Montague, avec des catégories syntaxiques relatives au contexte et aux domaines d'individus. Ce système peut être interprété dans la théorie constructive des types de Martin-Löf. Il est appliqué, d'abord, au français sans symboles mathématiques, avec une attention spéciale aux restrictions de sélection et aux dépendances par rapport à un contexte. Le fragment comprend des verbes et des adjectifs, des formes plurielles, des propositions relatives, et des syntagmes coordonnés. Ensuite, la grammaire est étendue au symbolisme mathématique et à son usage dans le texte français. Le fragment comprend des formules arithmétiques, la notation décimale, les conventions de parenthèses, les variables explicites, des énoncés de théorèmes et des structures textuelles de preuves. On finit par étudier quelques applications de la grammaire, basées sur l'implémentation déclarative de la grammaire dans ALF, un éditeur de preuves.*

SUMMARY — Grammatical Structure in Mathematical French: I.
A system of grammatical rules is presented to analyse a fragment of French that permits the expression of mathematical theorems and proofs. To this end, a version of Montague grammar is developed, with syntactic categories relativized to a context and to domains of individuals. This system can be interpreted in the constructive type theory of Martin-Löf. It is first applied to French without mathematical symbols, paying special attention to selectional restrictions and to dependencies on context. The fragment includes verbs and adjectives, plurals, relative clauses, and coordinated phrases of different categories. Second, the grammar is extended to mathematical symbolism and its embedding in French text. The fragment comprises arithmetical formulae, decimal notation, parenthesis conventions, explicit variables, statements of theorems, and textual structures of proofs. Finally, some applications of the grammar are studied, based on a declarative implementation in the proof editor ALF.

INTRODUCTION

Le langage mathématique peut être décrit par deux types de règles. Les règles de grammaire, dans le sens traditionnel, concernent le texte mathématique comme n'importe quel texte : sa construction grammaticale doit être correcte. Mais la correction grammaticale, dans le sens traditionnel, ne suffit pas pour qu'un texte mathématique soit complètement correct. Il lui faut, aussi, satisfaire à certaines règles mathématiques. Par exemple, si on

[1] Département de Philosophie, B.P. 24, 00014 Université d'Helsinki, Finlande.
Cet article est une synthèse du travail dont j'ai eu l'occasion de présenter des extraits à Paris et à Nancy en février 1996. Je suis reconnaissant aux publics de ces conférences pour leurs remarques et, en particulier, à M. Bourdeau et J.-P. Desclés, de l'Université de Paris-Sorbonne, à D. Lacombe, de l'Université de Paris VII, et à C. Retoré, de l'INRIA Lorraine, pour les entretiens privés sur les thèmes de l'article. P. Martin-Löf, de l'Université de Stockholm, m'a aidé pendant tout le travail par ses critiques et conseils. U. Egli, de l'Université de Constance, A. Lecomte, de l'Université de Grenoble et P. Mäenpää, de l'Université d'Helsinki ont lu la première version de l'article et fait des remarques pertinentes sur le contenu. En outre, M. Bourdeau et A. Lecomte ont corrigé le français des versions préliminaires. Les fautes qui restent dans la version finale sont miennes.

parle de *la fonction inverse de f*, il faut que *f* soit bijective. Autrement, l'expression *la fonction inverse de f* ne désigne aucun objet mathématique.

Ces conditions mathématiques peuvent être caractérisées comme des conditions sémantiques: elles concernent la signification des expressions plutôt que leur structure grammaticale. Mais si on accepte l'objectif classique de la grammaire, qui est de décrire les expressions de la langue comme signifiantes — ou, en termes modernes, la langue comme un système de communication — on voit qu'il y a des conditions sémantiques qui concernent la construction grammaticale. Un nom doit désigner un objet individuel, une phrase doit exprimer une proposition, etc. — Une expression qui ne désigne aucun objet du type qu'il lui faudrait est mal formée.

C'est surtout dans la linguistique formelle que les questions de signification sont parfois séparées de la grammaire, qui est conçue comme «syntaxe pure», pour être étudiées plus tard, dans un «composant sémantique». La grammaire de Montague est une exception remarquable.[1] Selon Montague, une telle approche n'est guère possible, parce qu'une syntaxe construite sans considérations sémantiques ignore des distinctions essentielles: les règles choisies à cause de leur simplicité superficielle sont trop simples pour qu'une sémantique soit définissable pour elles.[2] Une leçon importante de Montague est que l'introduction de la sémantique dans la grammaire ne la rend pas plus vague, mais, au contraire, la force d'être plus rigoureuse et plus explicite.

Même la grammaire de Montague est insuffisante pour notre objectif d'analyser le langage mathématique, surtout pour la raison qu'elle est fondée sur la théorie simple des types, qui n'est pas capable d'exprimer tout le contenu mathématique que nous voulons exprimer. C'est la **théorie constructive des types**, développée par Martin-Löf depuis les années 70,[3] qui a ouvert la possibilité d'une analyse plus exacte. Les questions qui peuvent être étudiées dans cette théorie, mais non dans la théorie simple des types, incluent les restrictions de sélection, la dépendance des expressions par rapport à un contexte, et l'interprétation des textes de démonstration.

Les grammaires logiques sont habituellement formulées comme des théories de la linguistique générale, c.-à-d. sans se borner à la langue mathématique.[4] Mais si une grammaire logique est possible — ce qui est souvent mis en question par ceux qui pensent que la langue naturelle «n'est pas logique» — elle devrait l'être pour le fragment mathématique de la langue, parce que c'est le domaine pour lequel la logique moderne a été développée originellement. Toutes ces questions d'ontologie, de pragmatique, et d'ambiguïté, qui sont tellement troublantes quand on étudie la langue dans sa totalité, peuvent être bien appréhendées dans la langue mathématique.[5] On n'a pas besoin de faire des idéalisations tout le temps: on peut donner une représentation fidèle de la langue elle-même.[6]

---

[1] Cf. la collection Montague (1974), ainsi que Chambreuil (1989).

[2] Cf. Montague (1974, p. 210).

[3] Cf. Martin-Löf (1984).

[4] L'auteur de cet article n'est pas une exception. Ainsi Ranta (1994) puise ses exemples linguistiques dans le corpus du «langage quotidien» familier de beaucoup d'ouvrages linguistiques et philosophiques. Mais les articles Ranta (1995, 1996) ont l'anglais mathématique comme leur thème. Le fragment de l'anglais analysé dans ces articles-là est, cependant, plus limité que le fragment du français que nous allons discuter dans cet article-ci.

[5] Par exemple, il n'est pas évident que les objets de la vie quotidienne forment des ensembles, dans le sens mathématique, ce qui serait, pourtant, nécessaire pour que la quantification sur ces objets soit bien définie.

[6] Si on veut donner une représentation fidèle d'une phrase telle que *Jean aime Marie*, on doit expliquer la référence contextuelle des noms *Jean* et *Marie* — parce qu'il n'y a pas de personnes uniques qui

Toutefois, il nous semble que la recherche sur la langue mathématique est intéressante aussi du point de vue de la linguistique générale, comme une sorte de travail de laboratoire, où on peut se concentrer sur des aspects bien définis et isolés. Si la langue, dans sa totalité, n'est pas logique, une étude partant de ce qui est logique montrera avec une grande précision où les limites se trouvent.

Nous n'avons pas rencontré beaucoup de recherche sur la langue informelle des mathématiques. Une référence importante sont les œuvres de de Bruijn, depuis les années 60. Dans ses articles et dans ses conférences non publiées, il y a beaucoup d'observations sur la structure du texte mathématique. La synthèse de ce travail n'est pas, pourtant, une étude linguistique, mais un langage artificiel, le vernaculaire mathématique (*mathematical vernacular*), qui s'approche aussi près de la langue naturelle que possible sans compromettre l'univocité.[7]

Une grande part de la recherche en théorie des types concerne le développement des systèmes interactifs de preuve, tels que ALF, Coq et LEGO.[8] Puisque les preuves produites par ces systèmes sont exprimées en langage formel, elles ne sont lisibles que par des spécialistes de la théorie des types. Ainsi on a étudié la possibilité de traduire les démonstrations formelles en langue naturelle. De tels programmes pour l'anglais et pour le français ont été réalisés comme fonctions d'interface des systèmes de preuve.[9] Une application possible de notre travail est un programme semblable, mais nous envisageons aussi la possibilité de traduire les textes informels en théorie des types: en effet, la grammaire que nous présenterons est symétrique par rapport de ces deux tâches. En tout cas, la précision nécessitée par les applications computationnelles sera un critère sous-entendu de notre grammaire. L'implémentation se réalise d'une façon naturelle par l'utilisation de la théorie des types comme langage de programmation.[10]

Tout en visant à la précision sémantique, nous ne voulons pas compromettre l'exactitude de la morphologie, ni nous contenter d'une langue très réduite, ou minimale pour les besoins d'expression. L'objectif est d'analyser toutes les combinaisons naturelles des catégories grammaticales considérées. En conséquence, notre grammaire pourra connaître plusieurs milliers d'expressions synonymes pour une proposition de quelque complexité.

Nous ne pensons pas que nous puissions dire quelque chose de nouveau sur les grands problèmes de la structure syntagmatique, qui jouent un rôle majeur dans la recherche en syntaxe formelle. Nous nous concentrerons plutôt sur les structures logiques qui échappent complètement à la grammaire syntagmatique. En même temps, notre analyse des structures syntagmatiques s'approche de la grammaire traditionnelle, ou bien de la grammaire de Montague. Les règles plus généralisantes des syntaxes contemporaines, telles que la grammaire syntagmatique généralisée, nous semblent, tout simplement, trop difficiles à combiner avec une sémantique assez détaillée et fiable.

La structure de l'article est la suivante: la section 1 discute quelques questions générales concernant la recherche grammaticale. La section 2 étudie le français général, c.-à-d. sans symboles ou autres structures spécifiquement mathématiques. La section 3 présente des règles d'usage du symbolisme mathématique dans le texte informel. La

---

s'appellent ainsi — la référence implicite qu'elle fait au temps, et l'ambiguïté potentielle du verbe *aimer*. L'analyse de la phrase 3 *est supérieur à* 2 est beaucoup plus facile à compléter.

[7]Cf. la collection de Bruijn (1994). Le système antérieur et mieux connu de de Bruijn, Automath, introduit plusieur structures désormais centrales de la théorie des types, dont les types dépendants et les objets-preuves.

[8]V. les volumes 806 et 996 de *Springer Lecture Notes in Computer Science*.

[9]Cf. Coscoy, Kahn et Théry (1994).

[10]Cf. Nordström *et al.* (1990).

section 4 contient des exemples et des applications pratiques.

Cet article ne contient pas d'introduction à la théorie constructive des types, pour laquelle il y a plusieurs textes facilement accessibles. Cependant, le reste de cette introduction contient des définitions de caractère général, qui sont utilisées dans les règles grammaticales. Ces sections peuvent être sautées, puis consultées après coup au besoin.

Les opérations logiques

Pour les opérations mathématiques et logiques, nous suivons la notation mathématique conventionnelle et, s'il n'y en a pas, la notation de Martin-Löf (1984). Toutefois, le typage des opérateurs syntaxiques et de leurs interprétations nécessite la théorie des types dite du **niveau supérieur**, qui n'y est pas présentée. Ainsi nous suivons la notation de Ranta (1994), qui est, en ce qui concerne la théorie du niveau inférieur, compatible avec Martin-Löf (1984). Il y a une exception importante: pour la **substitution** des termes $a_1, \ldots, a_n$ aux variables $x_1, \ldots, x_n$ dans un terme c, nous écrivons

$$c(x_1 = a_1, \ldots, x_n = a_n),$$

au lieu de $c(a_1, \ldots, a_n)$ et $c(a_1/x_1, \ldots, a_n/x_n)$, qui sont les notations employées dans Martin-Löf (1984) et Ranta (1994). La nouvelle notation est celle du **calcul des substitutions**. Cette extension récente de la théorie des types est indispensable à notre travail en formalisant la notation du contexte, et nous allons en résumer les parties utilisées dans la section suivante. Dans cette section-ci, nous résumons les définitions des opérations logiques.

Les règles du niveau supérieur comprennent les règles de la formation des types, qui sont le type des ensembles, dont chacun est lui-même un type, ainsi que les types de fonctions, où le type d'arrivée peut dépendre de l'argument de la fonction:

$$\text{set} : \text{type}, \qquad \frac{A : \text{set}}{A : \text{type}}, \qquad \frac{\begin{array}{c}(x : \alpha) \\ \alpha : \text{type} \quad \beta : \text{type}\end{array}}{(x : \alpha)\beta : \text{type}}.$$

Le type des propositions est identifié avec le type des ensembles. Le type des fonctions sans dépendence est défini comme un cas spécial:

$$\begin{aligned} &\text{prop} = \text{set}, \\ &(\alpha)\beta = (x : \alpha)\beta \text{ pour } \alpha, \beta : \text{type}. \end{aligned}$$

Les règles générales de la construction des objets sont l'application et l'abstraction:

$$\frac{c : (x : \alpha)\beta \quad a : \alpha}{c(a) : \beta(x = a)}, \qquad \frac{\begin{array}{c}(x : \alpha) \\ b : \beta\end{array}}{(x)b : (x : \alpha)\beta}.$$

Ces opérations sont reliées par deux règles de conversion, β et η:

$$\begin{aligned} &((x)b)(a) = b(x = a) \text{ pour } b : \beta\ (x : \alpha), a : \alpha, \\ &c = (x)(c(x)) \text{ pour } c : (x : \alpha)\beta. \end{aligned}$$

Ces règles ne connaissent que des fonctions à une place, mais les fonctions à plusieurs places sont introduites par la convention notationnelle

$$f(a_1, \ldots, a_n) \;=\; f(a_1) \ldots (a_n).$$

La théorie du niveau inférieur — comme n'importe quelle théorie mathématique — peut être introduite par des jugements du niveau supérieur, sans ajouter de règles d'inférence. Les opérateurs logiques utilisés dans cet article sont définis à partir des opérateurs

Σ : $(X$: set$)((X)$set$)$set,
Π : $(X$: set$)((X)$set$)$set,
$+$: (set)(set)set,
$\bot$: set,

par les équations

$(\exists x : A)B(x) = \Sigma(A, B)$ pour A : set, B : (A)prop,
$(\forall x : A)B(x) = \Pi(A, B)$ pour A : set, B : (A)prop,
$A \,\&\, B = \Sigma(A, (x)B)$ pour A, B : prop,
$A \supset B = \Pi(A, (x)B)$ pour A, B : prop,
$A \vee B = +(A, B)$ pour A, B : prop,
$\sim A = \Pi(A, (x)\bot)$ pour A : prop.

Par exemple, la formule

$$(\forall x : N)(\exists y : N)(\mathrm{Div}(x, y) \,\&\, \sim \mathrm{Pr}(y))$$

est écrite, dans sa forme primitive,

$$\Pi(N)((x)\Sigma(N)((y)\Sigma(\mathrm{Div}(x)(y))((z)\Pi(\mathrm{Pr}(y))((u)\bot)))).$$

La sémantique des constantes logiques est définie par les formes canoniques de leurs preuves, qui correspondent aux règles d'introduction :

pair : $(X$: set$)(Y$: (X)set$)(x$: $X)(Y(x))\Sigma(X, Y)$,
λ : $(X$: set$)(Y$: (X)set$)((x : X)Y(x))\Pi(X, Y)$,
i : $(X$: set$)(Y$: set$)(X)(X + Y)$,
j : $(X$: set$)(Y$: set$)(Y)(X + Y)$.

L'ensemble $\bot$ n'a pas de règles d'introduction, ce qui le définit comme l'ensemble vide. La notation conventionnelle du niveau inférieur est définie par les équations

$$(a, b) = \mathrm{pair}(A, B, a, b),\ (\lambda x)b(x) = \lambda(A, B, b),\ i(a) = i(A, B, a),\ j(b) = j(A, B, b),$$

qui suppriment les arguments de typage.

Les règles d'élimination correspondent à des constantes non-canoniques, qui sont définies par l'induction structurale sur les arguments canoniques. Nous en avons employées les projections gauche et droite ainsi que l'application,[11]

[11] Pour les autres règles d'élimination, par les constantes E, D et R_0, qui ne sont employées que dans la section 3.7, nous renvoyons aux œuvres de référence citées ci-dessus.

$$p : (X : \text{set})(Y : (X)\text{set})(\Sigma(X,Y))X,$$
$$p(A,B,\text{pair}(A,B,a,b)) = a,$$
$$q : (X : \text{set})(Y : (X)\text{set})(z : \Sigma(X,Y))Y(p(X,Y,z)),$$
$$q(A,B,\text{pair}(A,B,a,b)) = b,$$
$$\text{ap} : (X : \text{set})(Y : (X)\text{set})(\Pi(X,Y))(x : X)Y(x),$$
$$\text{ap}(A,B,\lambda(A,B,b),a) = b(a),$$

dans les formes définies par les équations

$$p(c) = p(A,B,c),\ q(c) = q(A,B,c),\ \text{ap}(c,a) = \text{ap}(A,B,c,a).$$

Les nombres naturels canoniques sont 0 et les successeurs,

$$0 : N,\ s : (N)N.$$

La règle d'élimination, c.-à-d. d'induction mathématique,

$$R : (Z : (N)\text{set})(z : N)(u : C(0))(v : (x : N)(y : Z(x))C(s(x)))C(z),$$
$$R(C,0,d,e) = d,$$
$$R(C,s(n),d,e) = e(n,R(C,n,d,e)),$$

permet la définition explicite des opérations arithmétiques telles que l'addition et la multiplication,

$$m+n = R((x)N,n,m,(x)(y)s(y)),\ m*n = R((x)N,n,0,(x)(y)(y+m)).$$

Les chiffres $1,2,3,\ldots$ sont utilisés comme abréviations des nombres canoniques ; la grammaire de cette convention n'est pas tout à fait simple, comme montré par la section 3.3.

Bien entendu, la notation bidimensionnelle des règles est beaucoup plus facile pour le lecteur humain que les jugements du niveau supérieur, et nous l'employons dans toutes les règles grammaticales. La transformation d'une règle en un jugement et vice-versa sera évidente : les prémisses correspondent aux types de départ et la conclusion au type d'arrivée. Par exemple, le jugement introduisant l'opérateur pair est exprimable par la règle

$$\frac{A : \text{set} \quad B : (A)\text{set} \quad a : A \quad b : B(a)}{\text{pair}(A,B,a,b) : \Sigma(A,B)}$$

Dans quelques cas, il est utile de traiter les catégories syntaxiques, ainsi que le type set (et prop, qui est identifié avec lui) comme des systèmes définis par induction. Il serait possible de formaliser cela par codage dans des univers. Nous ne le faisons pas dans le texte, parce que cela rendrait la notation très lourde, mais nous rappelons cette précondition quand elle est supposée, et renvoyons le lecteur intéressé au code de l'implémentation en ALF (cf. section 4.1), où le codage est explicite.

Les contextes

Pour les contextes dans la théorie des types, nous utilisons une notation présentée par Martin-Löf dans une série de conférences sur le **calcul des substitutions** en 1992. Cette notation diffère des notations mieux connues, publiées par Tasistro (1993) et Magnusson (1994). Nous écrivons

$$\Gamma : \text{cont},$$

pour dire que Γ est un **contexte**, une séquence d'**hypothèses**, de la forme

$$x_1 : A_1, \ \ldots, \ x_n : A_n,$$

où $n = 0, 1, 2, \ldots$ et chaque A_k est un ensemble qui dépend des hypothèses jusqu'au $n = k - 1$. Nous écrivons

$$J/\Gamma$$

pour un **jugement J dans le contexte** Γ, où J a n'importe quelle des quatre formes de jugement de Martin-Löf (1984).

Les contextes peuvent être engendrés d'une manière inductive, partant du **contexte vide**, (), par l'addition d'hypothèses,

$$() : \text{cont}, \qquad \frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma}{(\Gamma, \ x : A) : \text{cont}}.$$

Dans la seconde règle, x doit être une variable fraîche. Pour éviter l'accumulation de parenthèses, nous suivons les conventions d'écrire

$\Gamma, \ x : A$ au lieu de $(\Gamma, \ x : A)$,

$x : A$ au lieu de $(), \ x : A$,

ce qui explique la notation moins formelle pour les contextes dans la théorie des types avant le calcul des substitutions.

Les **objets donnés dans un contexte** comprennent, comme la base, toutes les variables du contexte:

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma}{x : A/\Gamma, \ x : A}.$$

Toutes les opérations de la théorie des types sont applicables dans n'importe quel contexte. Par exemple, la formation de la disjonction est valide dans la forme

$$\frac{A : \text{prop}/\Gamma \quad B : \text{prop}/\Gamma}{A \vee B : \text{prop}/\Gamma}.$$

Enfin, si le contexte Δ est une **extension** du contexte Γ, c.-à-d. si Δ contient toutes les hypothèses du Γ — nous pouvons désigner ce fait par $\Gamma \preceq \Delta$ — alors tout jugement dans Γ est aussi valide dans Δ,

$$\frac{J/\Gamma \quad \Gamma \preceq \Delta}{J/\Delta}.$$

Le **contexte dans un contexte**,

$$\Delta : \text{cont}/\Gamma,$$

est une séquence d'hypothèses qui peuvent dépendre, non seulement des hypothèses antérieures, mais aussi de celles de Γ. Il devrait être facile de se former une image intuitive de cette structure, mais parce qu'elle n'est pas présentée dans la littérature, nous lui consacrons quelques lignes. La définition inductive part du contexte vide, mais l'extension doit être définie simultanément avec l'**addition** des contextes,

$$() : \text{cont}/\Gamma, \qquad \frac{\Gamma : \text{cont} \quad \Delta : \text{cont}/\Gamma}{\Gamma + \Delta : \text{cont}}, \qquad \frac{\Delta : \text{cont}/\Gamma \quad A : \text{set}/\Gamma + \Delta}{(\Delta, \ x : A) : \text{cont}/\Gamma}.$$

L'addition est définie par les équations

$$\Gamma + () = \Gamma,$$
$$\Gamma + (\Delta,\ x : A) = (\Gamma + \Delta),\ x : A.$$

Enfin, nous avons besoin de l'addition des contextes dans un contexte,

$$\frac{\Gamma : \mathrm{cont} \quad \Delta : \mathrm{cont}/\Gamma \quad \Theta : \mathrm{cont}/\Gamma + \Delta}{\Delta + \Theta : \mathrm{cont}/\Gamma},$$

définie par les équations

$$\Delta + () = \Delta,$$
$$\Delta + (\Theta,\ x : A) = (\Delta + \Theta),\ x : A.$$

Mais la seconde équation n'est pas bien formée sans la règle de l'associativité de l'addition,

$$\frac{\Gamma : \mathrm{cont} \quad \Delta : \mathrm{cont}/\Gamma \quad \Theta : \mathrm{cont}/\Gamma + \Delta}{(\Gamma + \Delta) + \Theta = \Gamma + (\Delta + \Theta) : \mathrm{cont}},$$

qui, à son tour, est justifié par l'induction sur la structure de l'addition : pour l'addition vide,

$$(\Gamma + \Delta) + () = \Gamma + \Delta = \Gamma + (\Delta + ());$$

pour l'addition étendue, supposons $(\Gamma + \Delta) + \Theta = \Gamma + (\Delta + \Theta)$; alors

$$(\Gamma + \Delta) + (\Theta,\ x : A) = ((\Gamma + \Delta) + \Theta),\ x : A = (\Gamma + (\Delta + \Theta)),\ x : A$$
$$= \Gamma + ((\Delta + \Theta),\ x : A) = \Gamma + (\Delta + (\Theta,\ x : A)),$$

où chaque pas est une instance des équations définissantes, sauf le deuxième, qui utilise l'hypothèse d'induction.

La quantification sur les nouvelles hypothèses ajoutées à un contexte retourne une proposition dans le contexte originel :

$$\frac{\Gamma : \mathrm{cont} \quad \Delta : \mathrm{cont}/\Gamma \quad C : \mathrm{prop}/\Gamma + \Delta}{\begin{array}{l}\forall(\Delta, C) : \mathrm{prop}/\Gamma \\ \forall((), C) = C \\ \forall((\Delta,\ x : A), C) = \forall(\Delta, (\forall x : A)C)\end{array}}$$

$\exists(\Delta, C)$ est analogue.

Toutes les opérations définies ici peuvent être facilement modifiées pour s'appliquer aux extensions explicites (cf. section 3.2).

Les opérations sur les listes

Les listes habituelles de la théorie des types sont basées sur la liste vide et croissent vers la gauche. Les nôtres ont au moins un élément et croissent vers la droite :

$$\frac{A : \mathrm{set}}{\mathrm{list}_1(A) : \mathrm{set}}, \qquad \frac{a : A}{\mathrm{bs}_1(a) : \mathrm{list}_1(A)}, \qquad \frac{l : \mathrm{list}_1(A) \quad a : A}{\mathrm{cons}_1(l, a) : \mathrm{list}_1(A)}.$$

L'application d'une fonction aux éléments d'une liste est définie comme d'habitude,

$$\frac{f : (A)B \quad l : \mathrm{list}_1(A)}{\begin{array}{l}\mathrm{map}_1(f, l) : \mathrm{list}_1(B) \\ \mathrm{map}_1(f, \mathrm{bs}_1(a)) = \mathrm{bs}_1(f(a)) \\ \mathrm{map}_1(f, \mathrm{cons}_1(l, a)) = \mathrm{cons}_1(\mathrm{map}_1(f, l), f(a))\end{array}}$$

L'ensemble $\mathrm{list}_2(A)$ des listes d'au moins deux éléments et les opérations correspondantes, bs_2, cons_2 et map_2, sont définis d'une manière analogue. Puisque toute liste d'au moins deux éléments est représentable comme une liste d'au moins un élément, toutes les opérations définies pour list_1 s'appliquent aussi à list_2. L'ensemble $\mathrm{list}_3(A)$ des listes d'au moins trois éléments est analogue. La notation énumérative

$$[a, b, \ldots, c]$$

est employée pour tous les types de listes.

L'application d'un connecteur à une liste de propositions — conçues comme des éléments d'un univers, pour que la liste puisse être formée — est définie par récurrence pour C : (prop)(prop)prop, L : list_1(prop) :

$$\begin{aligned} &\mathrm{listconn}(C, \mathrm{bs}_1(A)) \;=\; A, \\ &\mathrm{listconn}(C, \mathrm{cons}_1(L, A)) \;=\; C(\mathrm{listconn}(C, L), A). \end{aligned}$$

La quantification sur les éléments d'une liste est définie d'une manière évidente pour A : set, B : (A)prop, l : $\mathrm{list}_1(A)$:

$$(\forall x \in l)B(x) \;=\; \mathrm{listconn}(\&, \mathrm{map}_1((x)B(x), l)).$$

Pour A : set, C : $(A)(A)$prop, l : $\mathrm{list}_2(A)$, nous définissons par récurrence

$$\begin{aligned} &(\forall x, y \in \mathrm{bs}_2(a, b))C(x, y) \;=\; C(a, b) \\ &(\forall x, y \in \mathrm{cons}_2(l, a))C(x, y) \;=\; (\forall x, y \in l)C(x, y) \;\&\; (\forall x \in l)C(x, a) \end{aligned}$$

La proposition

$$(\forall x, y \in l)C(x, y)$$

est donc la conjonction de toutes les propositions $C(x, y)$ à une direction ; un élément n'est pas relié à lui-même sauf si la liste le répète. La clôture symétrique serait exprimée par

$$(\forall x, y \in l)(C(x, y) \;\&\; C(y, x)),$$

mais dans une application typique, la relation C est déjà symétrique en soi.

Dans la grammaire des déclarations (section 3.5), nous avons besoin des contextes étendus par des listes de variables du même type. Cette opération est définie, pour Γ : cont, l : list_1(symb), A : set/Γ, par les équations

$$\begin{aligned} &\overline{x}_{\mathrm{bs}_1(t)} : A \;=\; x_t : A, \\ &\overline{x}_{\mathrm{cons}_1(l,t)} : A \;=\; (\overline{x}_l : A,\ x_t : A), \end{aligned}$$

où x doit être fraîche relative à Γ étendu par $\overline{x}_l : A$.

Les opérations sur les chaînes

À un certain niveau de l'implémentation, les chaînes doivent être définies comme des listes de caractères, pour qu'il soit possible de définir les opérations sur les chaînes, à partir de la concaténation. Nous présupposons simplement ce niveau, et exprimons la concaténation par la juxtaposition des matières graphiques.[12]

[12]Dans l'implémentation, nous représentons les chaînes par code LaTeX (Lamport 1986), qui permet le codage linéaire des notations bidimensionnelles telles que les exposants.

Les définitions présentées dans cette section sont communes à plusieurs langues. D'autres, qui sont spéciales au français, sont trouvées dans la section 2.1.

Une liste de chaînes est un objet plus articulé qu'une chaîne. Elle peut être **implosée** en une chaîne par la concaténation de tous ses éléments. Il y a des versions modifiées d'implosion : l'opération qui ajoute des blancs entre les éléments ; une autre qui y ajoute des virgule et des blancs ; une qui introduit une conjonction à l'intérieur de la dernière paire d'éléments ; et, finalement, une qui ajoute des points et met la majuscule à chaque élément.

$$\begin{aligned}
&\text{implode} : (\text{list}_1(\text{chaîne}))\text{chaîne}\\
&\text{implode}(\text{bs}_1(a)) = a\\
&\text{implode}(\text{cons}_1(l,a)) = \text{implode}(l)a\\
&\\
&\text{blancform} : (\text{list}_1(\text{chaîne}))\text{chaîne}\\
&\text{blancform}(\text{bs}_1(a)) = a\\
&\text{blancform}(\text{cons}_1(l,a)) = \text{blancform}(l)\ a\\
&\\
&\text{virgform} : (\text{list}_1(\text{chaîne}))\text{chaîne}\\
&\text{virgform}(\text{bs}_1(a)) = a\\
&\text{virgform}(\text{cons}_1(l,a)) = \text{virgform}(l),\ a\\
&\\
&\text{conjform} : (\text{chaîne})(\text{list}_1(\text{chaîne}))\text{chaîne}\\
&\text{conjform}(c,\text{bs}_1(a)) = a\\
&\text{conjform}(c,\text{cons}_1(l,a)) = \text{virgform}(l)\ c\ a\\
&\\
&\text{textform} = (l)\text{blancform}(\text{map}_1((x)(\underline{x}.),l)) : (\text{list}_1(\text{chaîne}))\text{chaîne}
\end{aligned}$$

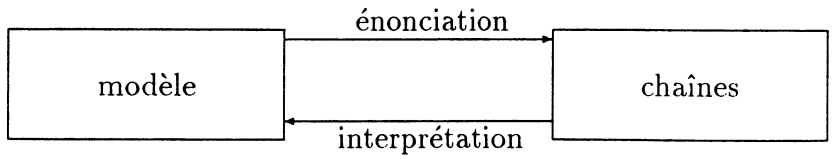
La présence d'une majuscule est désignée par soulignement.

1 CONSIDÉRATIONS GÉNÉRALES SUR LA GRAMMAIRE

1.1 *Le formalisme et la langue informelle*

Dans une grammaire logique, nous étudions les rapports entre deux systèmes. D'une part, nous avons une **langue informelle**, dont les expressions sont des énoncés non formalisés, ou, du point de vue mathématique, des **chaînes de caractères**, construites par l'opération de **concaténation**. De l'autre part, nous avons un **modèle**, un système d'**objets mathématiques**, tels que des ensembles, des éléments de ces ensembles, des propositions, et des fonctions individuelles ainsi que propositionnelles. L'opération qui associe un objet du modèle à une chaîne donnée est l'**interprétation**.

L'opération inverse — de trouver une expression linguistique pour un objet mathématique — n'a pas de nom établi, mais nous l'appellerons l'**énonciation**.[13] Ainsi nous pouvons dessiner un diagramme :



[13]Le terme *énonciation* doit être conçu comme dans les locutions « énoncer les faits », « l'énoncé d'un problème », et non comme « production individuelle d'une phrase ».

Dans l'enseignement de la logique formelle, on développe une faculté intuitive d'effectuer ces opérations. Dans la linguistique formelle, on cherche des règles précises pour le faire. Ces deux tâches sont plus générales que la méthode choisie ou qu'un modèle particulier : elles sont manifestées dans presque toute recherche linguistique. Ainsi le modèle peut être un système assez vaguement défini de « significations » ou d'« idées » ou de « concepts ». Parfois, on parle de la **sémasiologie**, qui « part des mots, du signifiant, pour en étudier la signification », et de l'**onomasiologie**, qui « part des concepts, des signifiés, pour voir comment la langue les exprime » (Grevisse, § 5).

Un énoncé informel n'a pas, en général, d'interprétation unique, mais peut en avoir plusieurs ou aucune. Dans le premier cas, l'énoncé est **ambigu** ; dans le second cas, il est **dénué de sens**. Par exemple, la chaîne

6 *est pair et divisible par* 4 *ou divisible par* 3

est ambiguë, parce qu'elle a les deux interprétations

$$(\mathrm{Pair}(6)\ \&\ \mathrm{Div}(6,4)) \vee \mathrm{Div}(6,3),$$
$$\mathrm{Pair}(6)\ \&\ (\mathrm{Div}(6,4) \vee \mathrm{Div}(6,3)),$$

tandis que la chaîne

6 *est impair par* 4

n'a pas de sens, parce que l'adjectif *impair* n'est défini que comme un prédicat à une place.

Il en est de même à propos de l'énonciation : typiquement, une proposition donnée peut être exprimée par plusieurs phrases informelles. Par exemple, la seconde des propositions ci-dessus est aussi exprimable par les phrases

6 *est pair et divisible par* 4 *ou par* 3,

6 *est pair et* 6 *est divisible par* 4 *ou par* 3,

qui sont donc **synonymes**. Il peut aussi y avoir des propositions qui sont si compliquées qu'il est impossible de trouver des phrases de la langue pour les exprimer. Par exemple, la proposition

$$(\forall \varepsilon > 0)(\exists \delta > 0)(\forall x, y)(|x - y| < \delta \supset \left|e^{3x} - e^{3y}\right| < \varepsilon)$$

est difficile à exprimer en français pur, sans l'aide du formalisme mathématique.

Il est donc assez clair que ni l'interprétation des chaînes ni l'expression des objets ne sont des fonctions, mais qu'elles sont deux faces d'une relation entre les objets et les chaînes. Cette relation,

la chaîne c exprime la proposition A,

peut être formalisée comme un prédicat à deux places,

$$\mathrm{Expr}(c, A)\ \ (c : \text{chaîne},\ A : \text{prop}).$$

Ce prédicat est la structure fondamentale de la grammaire **minimaliste**, dans sa version logique telle que conçue par Morrill (1994) : la grammaire est, exactement, un calcul qui

définit ce prédicat pour établir un rapport entre les chaînes et les objets du modèle. On n'emploie pas d'autres structures grammaticales que les chaînes et le modèle.[14]

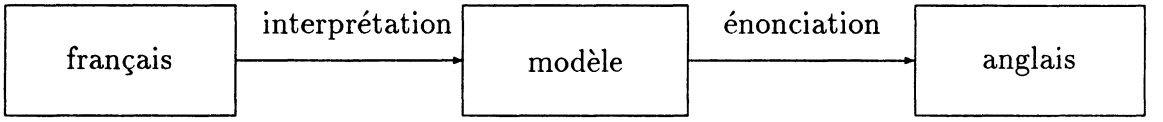
L'interprétation et l'énonciation sont ainsi des problèmes de recherche: pour interpréter une chaîne c, il faut trouver une proposition A telle que $\mathrm{Expr}(c, A)$ soit vraie. Pour exprimer une proposition A, il faut trouver une chaîne c telle que $\mathrm{Expr}(c, A)$. Ces problèmes peuvent être formalisés dans la théorie des types par la quantification existentielle:[15]

L'interprétation: $(\exists x : \mathrm{prop})\mathrm{Expr}(y, x)\ \ (y : \text{chaîne})$.

L'énonciation: $(\exists y : \text{chaîne})\mathrm{Expr}(y, x)\ \ (x : \mathrm{prop})$.[16]

Il est une propriété souhaitable de la grammaire que les solutions de ces problèmes soient effectives, mais elles n'appartiennent pas à la définition de la grammaire.

La grammaire minimaliste permet une définition naturelle de la **traduction** entre deux langues naturelles: pour traduire une phrase française en anglais, on l'interprète et, ensuite, on exprime la proposition obtenue comme une phrase anglaise.



Le modèle — ou, pour être précis, le formalisme mathématique dans lequel il est défini — fonctionne ici comme une **interlingua**, qui exprime les contenus sémantiques qui doivent être fidèlement transmis dans la traduction. Ainsi la préservation du contenu sémantique est garantie — pourvu que le modèle soit capable d'interpréter tout ce qu'on veut exprimer en français. Cette condition doit être une difficulté insurmontable pour la traduction de la langue générale par une interlingua: les formalismes les plus avancés de logique arrivent, avec peine, à exprimer tout le contenu des textes mathématiques, mais si on sort du fragment mathématique, il n'y a aucun formalisme comparable. Toutefois, dès qu'on trouve un formalisme sémantique pour le fragment pour lequel on veut construire un système de traduction, cette méthode est assez attirante.[17]

Formellement, la traduction peut être définie comme le problème de trouver une chaîne qui exprime la même proposition en anglais que la chaîne donnée exprime en

[14]Le prédicat $\mathrm{Expr}(c, A)$ présenté comme ici s'applique seulement aux chaînes interprétées comme des propositions, c.-à-d. aux phrases complètes. Pour analyser les expressions de toutes les catégories, on doit définir un prédicat à trois places, $\mathrm{Expr}(\alpha, c, a)$, où α est une catégorie, c est une chaîne, et a est un objet formel du type sémantique correspondant à α. C'est ainsi que fait Morrill, utilisant la notation $c - a : \alpha$.

[15]Pour être précis, le type prop dans ces définitions doit être borné à un univers U, qui est un ensemble défini par l'induction; cf. Martin-Löf (1984, pp. 87–91).

[16]Une opération générale d'interprétation produit une liste de propositions plûtot qu'une seule proposition,

$$\mathrm{list}((\exists x : \mathrm{prop})\mathrm{Expr}(x, y))\ \ (y : \text{chaîne}).$$

Si la liste est vide, la chaîne n'a pas de sens logique; si elle a plus qu'un élément, la chaîne est ambiguë. De la même façon, le problème de l'énonciation a une liste de solutions.

[17]On n'a pas de contrôle sur la structure grammaticale, parce que toute variation entre les expressions synonymes est supprimée dans l'interprétation — ainsi beaucoup de phrases françaises peuvent avoir la même traduction anglaise.

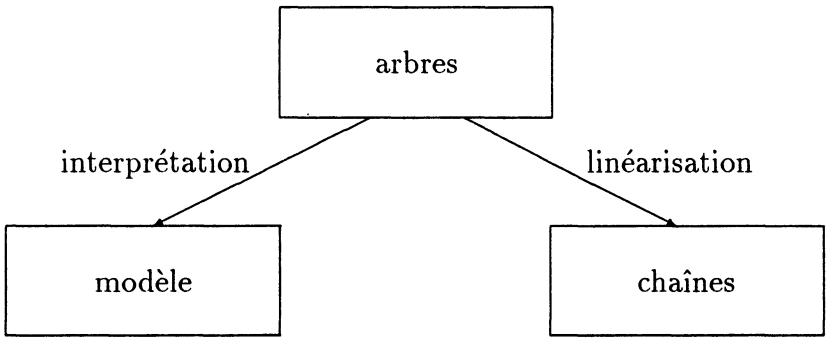
français

$$(\exists y : \text{chaîne})(\exists z : \text{prop})(\text{Expr}_f(x,z) \;\&\; \text{Expr}_e(y,z)) \;\; (x : \text{chaîne})$$

(les indices f, pour le français, et e, pour l'anglais, indiquent les deux versions du prédicat Expr).

1.2 *Les arbres syntaxiques*

Nous avons étudié une structure bipartite de la grammaire dans notre livre Ranta (1994), mais il est devenu évident qu'une structure avec un troisième élément, que nous appellerons les **arbres syntaxiques**, permet une description grammaticale beaucoup plus précise.[18] Un arbre syntaxique est une synthèse de toute l'information qu'il y a dans une expression française et dans son interprétation. Ainsi un arbre détermine et sa **linéarisation** en une chaîne, et son **interprétation** comme un objet du modèle:



Cette structure est la même que la structure de la grammaire de Montague (1974). Dans l'article «The Proper Treatment of Quantification in Ordinary English», les arbres syntaxiques s'appellent *analysis trees*; le modèle est un modèle de la logique intensionnelle basée sur la théorie simple des types. L'interprétation s'appelle *interpretation*, mais il n'y a pas de terme technique pour la linéarisation.[19]

L'arbre syntaxique est une synthèse de l'information sémantique, qui est nécessaire pour l'interprétation, et de l'information grammaticale, qui l'est pour la linéarisation. Pour analyser la phrase citée dans la section précédente,

$$6 \textit{ est pair et divisible par } 4 \textit{ ou divisible par } 3,$$

il y a donc deux arbres différents, que nous nommons A et B, correspondant aux deux interprétations possibles.[20] Désignant l'interprétation d'un arbre par une étoile, et sa linéarisation par un cercle, nous écrivons

$$A^\star = (\text{Pair}(6) \;\&\; \text{Div}(6,4)) \vee \text{Div}(6,3),$$
$$A^\circ = 6 \textit{ est pair et divisible par } 4 \textit{ ou divisible par } 3,$$
$$B^\star = \text{Pair}(6) \;\&\; (\text{Div}(6,4) \vee \text{Div}(6,3)),$$
$$B^\circ = 6 \textit{ est pair et divisible par } 4 \textit{ ou divisible par } 3,$$

[18] La grammaire de Ranta (1994) est plus minimale que celle de Morrill (1994), parce qu'elle ne connaît que les catégories logiques, sans distinctions syntaxiques; cf. section 1.8 ci-dessous.

[19] En anglais, il y a un mot argotique, *sugaring*, qui signifie l'opération inverse du *parsing*, et que nous avons trouvé très exact pour l'usage en grammaire. Ce mot n'est pas inconnu parmi les informaticiens français non plus.

[20] Les définitions des arbres A et B deviendront évidentes plus tard.

ce qui montre que la chaîne 6 *est pair et divisible par* 4 *ou divisible par* 3 est ambiguë. De la même façon, nous avons un arbre C avec les propriétés

$$C^\star = \text{Pair}(6)\ \&\ (\text{Div}(6,4) \vee \text{Div}(6,3)),$$
$$C^\circ = 6 \textit{ est pair et divisible par } 4 \textit{ ou par } 3,$$

ce qui montre, en combinaison avec les équations précédentes, qu'il y a au moins deux expressions françaises pour la proposition Pair(6) & (Div(6, 4) ∨ Div(6, 3)).

La linéarisation d'un arbre supprime l'information sémantique, tandis que son interprétation supprime l'information grammaticale. Cela signifie que les opérations inverses sont des problèmes de recherche — les problèmes de trouver un arbre avec une propriété désirée:

$$(\exists y : \text{arbre})(y^\circ = x)\ \ (x : \text{chaîne}),$$
$$(\exists y : \text{arbre})(y^\star = x)\ \ (x : \text{prop}).$$

Le premier problème est celui de l'**analyse** syntaxique (en anglais, du *parsing*): sa solution est un algorithme d'analyse. Puisque l'interprétation des arbres est une opération effective, l'interprétation d'une chaîne en logique peut être réduite au problème de l'analyse. De la même façon, l'expression d'une proposition logique est réduite au second problème. Le prédicat fondamental de la grammaire minimaliste est défini

$$\text{Expr}(x,y) = (\exists z : \text{arbre})(z^\star = y\ \&\ z^\circ = x).$$

Quel élément est l'**expression du français** dans une grammaire de cette forme? Il nous semble que c'est tantôt la chaîne — comme dans la grammaire minimaliste — tantôt l'arbre syntaxique. Par exemple, si on dit,

cuisines est le pluriel de *cuisine*,

la première expression du français, *cuisines*, est peut-être une chaîne, mais la seconde, *cuisine*, est certainement un arbre, puisqu'il n'y a pas d'opération de «former le pluriel» d'une chaîne, qui n'est que de la matière typographique (dans la langue parlée: de la matière phonique). Ce qui a une forme plurielle c'est un **objet grammatical**, un arbre. C'est comme objet grammatical, comme arbre de la catégorie des noms communs, que *cuisine* a pour pluriel *cuisines*. Et puis il y a un autre objet grammatical, qui se manifeste dans la phrase *il cuisine bien*, avec le pluriel *cuisinent*.[21]

Nous suivons la convention de désigner les chaînes en caractères italiques et les arbres en caractères romains. Ainsi il pourrait y avoir l'arbre

cuisine,

d'une catégorie nominale, dont la linéarisation produirait des chaînes telles que *cuisine, cuisines*, et l'arbre

cuisiner,

[21] Dans une grammaire minimaliste, on pourrait exprimer un tel rapport entre deux chaînes en leur associant la même signification. Mais cela n'est pas assez précis, s'il y a des mots synonymes dans la langue, parce que deux chaînes synonymes ne peuvent pas être distinguées l'une de l'autre par le prédicat Expr. Par exemple, si les chaînes *fonction* et *application* sont synonymes, toutes les deux ont la chaîne *fonctions* comme leur forme plurielle, dans le sens que celle-ci exprime la même chose au pluriel que celles-là expriment au singulier.

d'une catégorie verbale, dont la linéarisation produirait des chaînes telles que *cuisine, cuisines, cuisinons, cuisinent.*

Pour la traduction, la grammaire tripartite offre deux définitions naturelles. Il y a la définition déjà présentée pour la grammaire minimaliste, selon laquelle la traduction est l'opération composée de l'interprétation logique et de l'expression de cet objet dans l'autre langue. Mais la tâche générale est une traduction entre les arbres syntaxiques, avec la condition que l'interprétation soit préservée:

$$(\exists y : \text{arbre}_e)(y^\star = x^\star) \ (x : \text{arbre}_f).$$

Une solution du problème exprimé par cette formule peut être sensible à la structure grammaticale: les synonymes peuvent être traduits différemment. La traduction minimaliste est un cas spécial, qui supprime toutes les différences entre les chaînes synonymes.

1.3 *Les règles grammaticales*

L'arbre syntaxique est l'objet grammatical par excellence, l'objet qui porte les propriétés définitionnelles d'une expression linguistique — d'une part, qui a telle ou telle signification, et de l'autre, qui se manifeste en telles ou telles formes linéaires. La grammaire est une théorie de ces objets. Elle les définit en décrivant comment ils sont formés et, ensuite, comment ils sont interprétés et linéarisés.

Il n'y a pas d'ensemble universel des arbres, mais plusieurs **catégories syntaxiques**, telles que la phrase, le nom commun, le verbe transitif. Pour chaque catégorie α, il y a un **type sémantique** correspondant. Nous l'appelons l'**interprétation** de α et le désignerons par $\alpha^\star$.[22] C'est à ce type $\alpha^\star$ qu'appartient l'interprétation de chaque arbre de la catégorie α. Les linéarisations des arbres appartiennent au type des chaînes de caractères.

Ainsi la forme générale d'une règle grammaticale est

$$\frac{a_1 : \alpha_1 \ \cdots \ a_n : \alpha_n}{\begin{array}{l} C(a_1, \ldots, a_n) : \alpha \\ C(a_1, \ldots, a_n)^\star = F(a_1^\star, \ldots, a_n^\star) \\ C(a_1, \ldots, a_n)^\circ = G(a_1^\circ, \ldots, a_n^\circ) \end{array}}$$

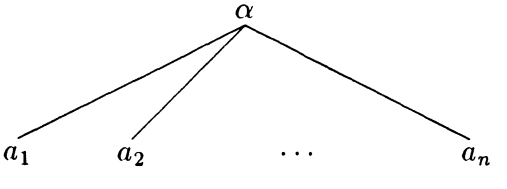
Cette forme correspond à la forme

$$\alpha \rightarrow \alpha_1 \ \ldots \ \alpha_n$$

d'une règle syntagmatique: les catégories des prémisses de notre règle correspondent aux catégories à la droite de la flèche, et la catégorie de la conclusion correspond à la catégorie à la gauche de la flèche. L'arbre formé,

$$C(a_1, a_2, \ldots, a_n),$$

correspond à l'arbre



[22] C'est la même chose que le *domain of possible denotations* chez Montague.

en notation graphique ; tous les deux contiennent la même information, avec l'exception qu'il peut y avoir, dans notre grammaire, une autre règle avec les mêmes prémisses et la conclusion

$$C'(a_1, a_2, \ldots, a_n) : \alpha,$$

dont la notation graphique serait indistinguable de la notation pour la forme C. Donc la notation des arbres par des termes fonctionnels est un peu plus expressive que la notation graphique.

Pour un exemple élémentaire, considérons la règle syntagmatique pour former une **phrase** (P) à partir d'un **syntagme nominal** (SN) et d'un **verbe à une place** (V1; aussi appelé **syntagme verbal**),

$$\mathrm{P} \rightarrow \mathrm{SN\ V1}.$$

Cette règle n'indique pas toute l'information appartenant à la phrase formée. Pour l'exprimer, donnons aux catégories utilisées les interprétations suivantes, que la grammaire de Montague a rendues assez familières :

$$\mathrm{P}^\star = \mathrm{prop},\ \mathrm{V1}^\star = (D)\mathrm{prop},\ \mathrm{SN}^\star = ((D)\mathrm{prop})\mathrm{prop}.$$

D est le domaine des individus, prop est le type des propositions. Ainsi les verbes sont interprétés comme des fonctions propositionnelles, et les syntagmes nominaux comme des fonctions propositionnelles sur les fonctions propositionnelles. Maintenant, nous pouvons formuler la règle

$$\frac{Q : \mathrm{SN} \quad F : \mathrm{V1}}{\begin{array}{l}\mathrm{PrédV1}(Q, F) : \mathrm{P} \\ \mathrm{PrédV1}(Q, F)^\star = Q^\star(F^\star) \\ \mathrm{PrédV1}(Q, F)^\circ = Q^\circ\, F^\circ\end{array}}$$

(Nous avons choisi le nom PrédV1 parce que c'est la règle de prédication avec un verbe à une place.)

La forme PrédV1 est très fréquemment la forme principale d'une phrase. Voici quelques exemples, où la structure est indiquée par les parenthèses :

(*la série harmonique*)(*diverge*),

(51 *et* 91)(*ne sont pas premiers*),

(*la ligne de connexion entre les points a et b*)(*est orthogonale à la droite l*).

Il est possible de définir des **macros**, des arbres abrégés. Par exemple, si nous introduisons la règle de complémentation pour le verbe à deux places,

$$\mathrm{V1} \rightarrow \mathrm{V2\ SN}$$

$$\frac{F : \mathrm{V2} \quad Q : \mathrm{SN}}{\begin{array}{l}\mathrm{ComplV2}(F, Q) : \mathrm{V1} \\ \mathrm{ComplV2}(F, Q)^\star = (x)Q^\star((y)F^\star(x, y)) \\ \mathrm{ComplV2}(F, Q)^\circ = F^\circ\, Q^\circ\end{array}}$$

(la catégorie V2 est interprétée $(D)(D)\mathrm{prop}$), nous pouvons définir, comme une macro, la prédication d'un verbe à deux places,

$$\frac{Q : \mathrm{SN} \quad F : \mathrm{V2} \quad Q' : \mathrm{SN}}{\mathrm{PrédV2}(Q, F, Q') = \mathrm{PrédV1}(Q, \mathrm{ComplV2}(F, Q')) : \mathrm{P}}$$

L'interprétation et la linéarisation de la macro sont des conséquences de sa définition :

$$\text{PrédV2}(Q, F, Q')^\star = Q^\star((x)Q'^\star((y)F^\star(x,y))),$$
$$\text{PrédV2}(Q, F, Q')^\circ = Q^\circ\ F^\circ\ Q'^\circ.$$

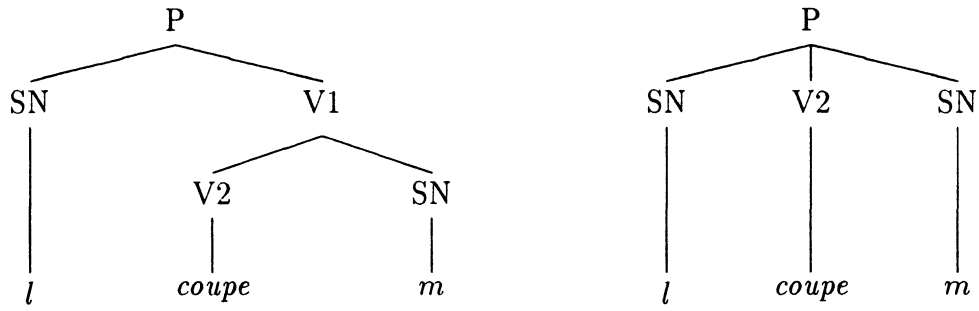
Dans la grammaire syntagmatique, l'introduction d'une macro correspond à l'introduction d'une nouvelle règle — pour notre exemple,

$$\text{P} \rightarrow \text{SN V2 SN}.$$

Si l'identité des arbres est conçue comme l'identité graphique, cette règle rend un énoncé tel que

l coupe m

ambigu : il y a deux arbres distincts qui l'analysent,



Mais dans notre interprétation des règles syntagmatiques, ces arbres sont égaux parce que l'arbre de droite n'est qu'une version abrégée de l'arbre de gauche.

Les arbres syntaxiques sont donc traités comme des véritables objets mathématiques, avec la distinction entre les formes **canoniques** et les macros, celles-ci définies par rapport aux formes canoniques. Les règles qui introduisent les formes canoniques jouent le rôle des axiomes de la théorie, tandis que les règles introduisant les macros en sont des théorèmes. Comme dans une théorie mathématique quelconque, nous pouvons renvoyer aux théorèmes sans descendre tout le temps jusqu'aux axiomes. Cela rendra les analyses grammaticales plus courtes que dans une grammaire avec la notion graphique d'arbre, et tout aussi exactes.

1.4 *Les traits morphologiques*

Nos règles de prédication et de complémentation ne sont pas encore assez exactes. Il y a de l'information et linguistique et logique qu'elles ignorent, et qu'il faut y ajouter pour obtenir des règles exactes. Quant à l'information linguistique, il nous faut incorporer les **traits morphologiques**, telles que le **genre**, le **nombre** et le **mode**. Ces catégories de traits peuvent être définies, simplement, comme des ensembles énumérés :

$$\text{genre} = \{\text{masc}, \text{fém}\},$$
$$\text{nombre} = \{\text{sg}, \text{pl}\},$$
$$\text{mode} = \{\text{ind}, \text{subj}\}.^{23}$$

[23] Il y a aussi la voix, le temps et la personne, mais à l'exception de la voix, ils ne sont guère employés dans la langue mathématique. Par exemple, la personne est presque toujours la troisième. La première personne apparaît dans les locutions telles que *nous avons une contradiction*, mais elle n'est jamais formée des prédicats mathématiques. La raison de l'omission du mode impératif est analogue.

Les traits morphologiques entrent dans les constructions syntaxiques dans deux rôles. Ils sont tantôt des **paramètres** dont dépend la forme qu'un arbre reçoit en linéarisation, tantôt des **traits inhérents** de l'arbre. Cette distinction est assez claire chez Grevisse, par exemple, dans les caractérisations suivantes :

> Le genre est une propriété du nom, qui le communique, par le phénomène de l'accord... au déterminant, à l'adjectif... (§ 454).
>
> Au contraire du genre, le nombre n'est pas un caractère du nom considéré en soi, mais il correspond aux besoins de la communication. La plupart des noms connaissent la variation en nombre... (§ 492).
>
> L'adjectif est un mot qui varie en genre et en nombre, genre et nombre qu'il reçoit, par le phénomène de l'accord... du nom auquel il se rapporte (§ 526).

Puisque notre système de catégories diffère du système de Grevisse, il nous faudra parfois modifier les caractérisations qu'il donne, mais il est clair qu'il s'agit de la même distinction.

Pour préciser la règle de prédication présentée dans la section précédente, il nous faut donc expliquer la morphologie des trois catégories qui y figurent. La phrase varie en mode, mode qui est déterminé par la construction dont la phrase fait partie. Par exemple, l'énoncé d'un théorème se met à l'indicatif, tandis qu'une phrase introduite par *il n'est pas vrai que...* se met au subjonctif. La règle

$$\frac{A : \mathrm{P} \quad m : \mathrm{mode}}{A^{\circ}(m) : \mathrm{chaîne}}$$

exprime le fait que la linéarisation d'une phrase dépend d'un paramètre de mode.

Le verbe varie en genre, en nombre et en mode. La variation en genre n'est connue par Grevisse qu'au participe (v. § 737), mais puisqu'un syntagme verbal peut être construit à partir d'un adjectif, au moyen d'une copule, il lui faut varier en genre aussi. Typiquement, le verbe reçoit son genre et son nombre par l'accord du sujet de la phrase, et son mode de la phrase où il sert de prédicat.

Le syntagme nominal ne varie ni en genre ni en nombre : ces traits lui sont inhérents. Nous avons les règles

$$\frac{Q : \mathrm{SN}}{\mathrm{gen}(Q) : \mathrm{genre}}, \qquad \frac{Q : \mathrm{SN}}{\mathrm{num}(Q) : \mathrm{nombre}}.$$

C'est précisément grâce à ces déterminations qu'on peut fixer, dans la règle de prédication, le genre et le nombre du verbe.[24]

Dans les langues qui ont des **cas**, le syntagme nominal a le paramètre de cas. En français, les fonctions des cas du latin ont été héritées par les **prépositions**, y incluse l'absence de préposition, qui correspond aux cas nominatif et accusatif. Dans la grammaire syntagmatique, les prépositions sont habituellement traitées comme des mots, d'une catégorie spéciale, Prép. Les formes prépositionnelles des syntagmes nominaux appartiennent à une catégorie spéciale, SP, des *syntagmes prépositionnels.* Les expressions de SP sont formées par la règle syntagmatique

$$\mathrm{SP} \rightarrow \mathrm{Prép\ SN}$$

[24]Comme nous verrons plus tard, le nom commun diffère du syntagme nominal — et du nom propre, qui en est un cas spécial — en variant en nombre.

(cf. Abeillé 1993, p. 57).

Mais la considération historique — qui, en soi, ne devrait rien signifier pour la grammaire du français contemporain — suggère l'introduction du **préfixe** comme un trait morphologique dans la grammaire. Un préfixe est soit une préposition soit **nul**, que nous désignerons par $-$, par le signe moins :

$$\text{préfixe} \;=\; \{-, \grave{a}, avec, de, entre, par, pour, sur\}$$

(nous n'aurons pas besoin d'autres prépositions).

Il y a deux arguments pour le traitement du préfixe comme un paramètre du syntagme nominal, au lieu de la séparation en syntagmes prépositionnels et syntagmes nominaux : 1° Quelques formes prépositionnelles, telles que la forme *du point* de *le point*, résultent d'une opération morphologique plus complexe que la concaténation. 2° Les verbes à un « complément indirect » (complément introduit par une préposition) pourront être combinés selon les mêmes règles que les verbes transitifs (dont les compléments sont introduits sans prépositions). Une catégorie, V2 des verbes à deux places, suffira pour les deux types de verbes.[25]

Maintenant, nous pouvons reformuler la règle de prédication avec l'information morphologique :

$$\frac{Q : \text{SN} \quad F : \text{V1}}{\begin{array}{l}\text{PrédV1}(Q,F) : \text{P} \\ \text{PrédV1}(Q,F)^\star \;=\; Q^\star(F^\star) \\ \text{PrédV1}(Q,F)^\circ(m) \;=\; Q^\circ(-)\; F^\circ(\text{gen}(Q), \text{num}(Q), m)\end{array}}$$

Dans une théorie contemporaine très influente, la grammaire d'unification, les traits morphologiques sont des arguments des catégories elles-mêmes. La règle de prédication, qui engendre les mêmes chaînes que notre règle, serait

$$\text{P}(m) \;\rightarrow\; \text{SN}(g,n)\; \text{V1}(g,n,m).$$

Il n'y a pas de distinction entre les paramètres et les traits inhérents : tous les deux sont simplement des arguments des catégories. Il n'y a pas, non plus, de dépendance entre les constituants de la phrase. La règle ne dit pas que le verbe s'accorde avec le sujet, mais plutôt que les deux s'accordent l'un avec l'autre : l'accord devient un rapport symétrique. Bien que cette technique puisse être suffisante pour l'engendrement des formes correctes, et que son implémentation sur ordinateur soit facile, nous préférons l'analyse basée sur la distinction traditionnelle. Or, nous ne voyons pas comment cette distinction pourrait être exprimée dans une grammaire minimaliste sans les arbres syntaxiques.[26]

1.5 *Les distinctions de domaine*

Nous avons montré l'interprétation de quelques catégories syntaxiques dans une théorie des types avec un seul domaine d'individus, D. La théorie constructive des types en

[25] Un argument contre le traitement du préfixe comme un trait morphologique serait qu'il n'est pas distribué aux constituants du syntagme, comme le cas : il n'y a pas d'accord de l'adjectif avec le nom en préfixe.

[26] Cf. Shieber (1986) et Abeillé (1993) pour le traitement de l'accord par l'unification. Dans la grammaire catégorielle, la même technique à été adopté par Morrill (1994), bien qu'il observe que l'unification « is only a computational operation, and we regard its promotion as a foundational principle of grammar as misleading » (p. 180).

connaît plusieurs : tout objet du type

set

des **ensembles** est lui-même un domaine. Nous pouvons ainsi généraliser toutes les catégories interprétées par référence au domaine D en les remplaçant par des **familles de catégories** dépendantes des variables du type set. S'il y a plusieurs références à D, chacune d'elles peut être un domaine distinct. Ainsi, nous avons les familles de catégories

V1(A)	interprétée	(A)prop
V2(A, B)	interprétée	(A)(B)prop
SN(A)	interprétée	((A)prop)prop

Les règles syntaxiques sont généralisées en conséquence. Par exemple, la règle de prédication devient

$$\frac{A : \text{set} \quad Q : \text{SN}(A) \quad F : \text{V1}(A)}{\begin{array}{l}\text{PrédV1}(A, Q, F) : \text{P} \\ \text{PrédV1}(A, Q, F)^\star = Q^\star(F^\star) \\ \text{PrédV1}(A, Q, F)^\circ(m) = Q^\circ(-)\ F^\circ(\text{gen}(Q), \text{num}(Q), m)\end{array}}$$

En ce cas-ci, nous pouvons formuler une règle syntagmatique qui montre les dépendances du domaine,

$$\text{P} \rightarrow \text{SN}(A)\ \text{V1}(A),$$

mais il y aura aussi des règles qui ne sont pas exprimables dans la notation de la grammaire syntagmatique.[27]

Chaque instance où les variables A et B sont remplacées par des domaines particuliers est une **catégorie syntaxique relative à des domaines**. Par exemple, la catégorie V1(N) contient les syntagmes verbaux définis pour les nombres naturels. La chaîne *être impair* est analysable comme une expression de cette catégorie, tandis que la chaîne *être vertical* ne l'est pas ; elle résulte à son tour d'un arbre du type V1(Ln) des verbes définis pour les droites. Puisque le chiffre 3 est analysable comme une expression du type SN(N) mais non du type SN(Ln), la règle de prédication admet la chaîne

3 *est impair*

mais n'admet pas la chaîne

3 *est vertical.*

En incorporant les dépendances du domaine dans les catégories et dans les règles, nous atteignons ce qu'on appelle les **restrictions de sélection** dans la linguistique générale. Une discussion pionnière se trouve dans les *Aspects* de Chomsky (1965), où un système rudimentaire de domaines est construit par des traits binaires tels que ±Abstrait, ±Animé et ±Humain (p. 82). Par exemple, *jouer* est un verbe à deux places, dont le sujet doit être +Animé, et cela explique pourquoi la chaîne

[27] Par exemple, la règle de détermination,

$$\text{SN} \rightarrow \text{Dét CN},$$

forme un syntagme nominal dont le domaine est l'interprétation du nom commun déterminé (cf. section 2.7). Cette dépendance n'est pas exprimable par une règle syntagmatique.

Jean joue au golf

est correcte tandis que

le golf joue à Jean

ne l'est pas.

Un grand problème pour une théorie des restrictions de sélection dans la langue entière est, naturellement, qu'on ne connaît aucun système de domaines qui s'applique à tout ce qui existe. Le système défini par des traits binaires puise son inspiration dans l'ontologie aristotélicienne, qui est bien développée dans certaines disciplines, dont la zoologie, mais qui est très problématique comme principe général. Donc dans les mathématiques, des systèmes beaucoup plus raffinés ont été développés, tels que la théorie des ensembles, et c'est précisément la théorie constructive des ensembles dont nous nous servons pour formaliser les restrictions de sélection. Évidemment, toute classification binaire est définissable dans la théorie des ensembles.

Même dans la langue mathématique, il y a une objection aux restrictions de domaine, à savoir, la possibilité d'utiliser la même expression pour plusieurs domaines. Ainsi on dit, non seulement qu'une droite coupe une autre droite, mais aussi qu'un cercle coupe un autre cercle, et qu'une droite coupe un cercle. Alors, il ne semble pas juste de catégoriser *couper* comme un verbe de la catégorie V2(Ln, Ln) ou d'une catégorie V2(A, B) quelconque avec les domaines A et B fixés.

Pour répondre à cette objection nous rappelons la distinction entre les deux notions d'expression, la chaîne et l'arbre. Une chaîne peut correspondre à plusieurs arbres distincts, dont elle est obtenue par linéarisation. Il peut donc y avoir toute une famille d'arbres, dont chacun produit la même chaîne. Un cas extrême sont les expressions **polymorphes**, qui sont usitées dans tous les domaines. L'adjectif *égal* est un exemple: il appartient, pour tout domaine A, à la catégorie A2($A, A,$ *à*) des adjectifs à deux places, dont le complément est introduit par la préposition *à*. Dans la théorie des types, il y a un prédicat correspondant,

$$I : (X : \text{set})(X)(X)\text{prop},$$

qui fournit l'interprétation à cet adjectif polymorphe:

$$\frac{A : \text{set}}{\begin{array}{l}\text{égal(A)} : \text{A2}(A, A, \grave{a}) \\ \text{égal(A)}^\star = I(A) \\ \text{égal(A)}^\circ = \textit{égal}\end{array}}$$

Ainsi nous avons des adjectifs-arbres telles que égal(N), égal(R), égal(Ln), qui produisent le même adjectif-chaîne *égal* dans la linéarisation.[28]

Une version restreinte de la polymorphie sont les expressions définies pour tous les domaines qui satisfont à une condition donnée. Ainsi l'adjectif *supérieur* est défini pour

[28]La forme d'un adjectif produite par la linéarisation dépend du genre et du nombre (cf. section 2.2). Ainsi la définition exacte de la linéarisation, dans ce cas-ci, serait

$$\text{égal(A)}^\circ(g, n) = \text{adj}_{al}(\textit{égal}, g, n),$$

où adj_{al} est une opération morphologique telle qu'expliquée dans la section 2.1.

tout domaine ordonné, c.-à-d. pour tout domaine A muni d'une relation R et d'une preuve c du fait que R est un ordre dans A:

$$\frac{A : \text{set} \quad R : (A)(A)\text{prop} \quad c : \text{Ord}(A, R)}{\begin{array}{l}\text{supérieur(A,R,c)} : \text{A2}(A, A, \grave{a}) \\ \text{supérieur(A,R,c)}^{\star} = R \\ \text{supérieur(A,R,c)}^{\circ} = \textit{supérieur}\end{array}}$$

Notons que la linéarisation supprime et le domaine A, et la relation R, et la preuve c. Si on veut parler des ordres distincts de A, on peut introduire l'adjectif *supérieur dans le sens...*, qui indique de quel ordre il est question.

Un autre type de pluralité de domaines est la **surcharge** d'un mot (en anglais informatique, *overloading*): par coïncidence, le mot a des usages divers. Ces usages peuvent être reliés par des associations extramathématiques, mais il n'y a pas de lien formel. Par exemple, l'adjectif *complet* est usité pour un espace métrique où toute suite de Cauchy converge, mais aussi pour une théorie formelle dans laquelle toute formule ou sa négation est démontrable. Les deux adjectifs-arbres distincts — qui peuvent être désignés complet_1 et complet_2 — produisent le même adjectif-chaîne dans la linéarisation.[29]

Habituellement, même la grammaire logique ignore les distinctions de domaine, pour la simple raison que la logique utilisée, la théorie simple des types, n'en fait pas. On pourrait essayer de justifier cette pratique par des exemples de typage «moins strict» dans le langage informel, mais nous trouvons que le typage strict, avec les concepts de polymorphie et de surcharge, donne une explication beaucoup plus articulée de tels exemples.

1.6 *La dépendance du contexte*

Pour analyser un texte mathématique, même très simple, il faut expliquer comment y évolue le **contexte**. Dans le cas idéal, le texte commence dans le contexte **vide**, mais il y a des structures grammaticales qui **étendent** le contexte — par exemple, par une déclaration d'une variable,

soit f une fonction réelle,

ou par une hypothèse,

supposons que f est dérivable en tout point de R.

La première phrase étend le contexte par la variable f, qui est alors utilisée comme un syntagme nominal dans la seconde phrase.

Dans la théorie constructive des types, l'hypothèse n'est qu'une déclaration d'une variable parmi d'autres, qui, dans ce cas, est un objet-preuve. Après les deux phrases ci-dessus, nous nous trouvons dans le contexte

$$f : R \to R,\ z : (\forall x : R)\text{Dér}(f, x),$$

dans lequel nous pouvons interpréter l'énoncé

[29]Un trait typique de la surcharge des mots est qu'elle varie d'une langue à l'autre. Par exemple, l'adjectif finnois *täydellinen* est usité et pour les théories, dans le sens *complet*, et pour les nombres entiers, dans le sens *parfait*. À cause de ce phénomène, il n'est pas possible de traduire d'une langue à l'autre sans une analyse qui va jusqu'aux distinctions de domaine.

si f est croissante, $f'(x)$ est supérieure à 0 pour tout x

comme la proposition

$$\mathrm{Cr}(f) \supset (\forall x : R)(D(f, x, \mathrm{ap}(z, x)) \geq 0).$$

Le rôle essentiel du contexte dans cette phrase — outre qu'il lui fournit les variables utilisées — est de fournir une preuve du fait que f est dérivable en x, qui est nécessaire pour la formation du terme $f'(x)$ comme une expression qui désigne un nombre réel: son interprétation est construite par l'opérateur

$$D : (f : R \to R)(x : R)(z : \mathrm{Dér}(f, x))R,$$

à trois places. Le troisième argument est supprimé dans la linéarisation: la notation mathématique informelle ne l'indique pas dans le terme lui-même. Cependant, quand la notation est introduite dans un manuel d'enseignement, la condition est mise en évidence, et un auteur soigneux qui écrit $f'(x)$ fournit à son lecteur l'information nécessaire pour que le terme soit bien formé.

Dans la langue usuelle, les variables explicites ne sont guère utilisées, mais les objets-preuves fournis par le contexte apparaissent dans le phénomène de la **présupposition**. Par exemple, la phrase

la nation belge adore son roi

est bien formée parce qu'il y a un roi des Belges; remplacez *belge* par *française*, et la phrase devient anomale.

Dans la linguistique générale, la présupposition n'appartient pas, habituellement, à la grammaire formelle, mais reste en marge comme un phénomène un peu vague. Elle est étudiée dans la «pragmatique», qui est l'étude non de la langue comme système mais de l'usage que les locuteurs en font. Cette solution nous semble arbitraire. Elle est peut-être imposée par les limitations des formalismes, qui n'ont aucun moyen pour l'exprimer. En tout cas, dans la langue mathématique, la présupposition appartient à la structure de la langue; elle ne laisse point de place pour les décisions pragmatiques des locuteurs.

L'**anaphore**, l'usage des pronoms et de l'article défini pour renvoyer à quelque chose qui est donnée dans le contexte, est mieux reconnu comme appartenant à la structure de la langue. L'étude des «donkey sentences», introduite par Geach (1962), occupe un rôle central dans la grammaire logique. Dans la langue mathématique, il est facile de trouver des exemples ayant la même structure: dans la phrase

si un nombre est pair, il est divisible par 2,

le pronom *il* renvoie au nombre introduit par l'antécédent.[30]

Habituellement, les expressions primaires dépendantes du contexte sont des termes singuliers. Un tel terme peut être utilisé comme un syntagme nominal, qui est le sujet d'une phrase ou le complément d'un verbe. Les expressions complexes d'à peu près toutes les catégories peuvent contenir des termes qui dépendent du contexte et, ainsi,

[30]Un problème concernant les pronoms est l'**unicité** de leur référence: il arrive souvent qu'il y ait plusieurs interprétations possibles fournies par le contexte, et que le choix soit fait à partir de considérations assez compliquées (cf. Ranta 1994, chapitre 4, où quelques principes sont discutés). Dans un texte mathématique, on préfère les variables explicites aux pronoms, s'il y a un risque de difficulté d'interprétation. Nous omettons la discussion des pronoms dans cet article.

elles peuvent dépendre elles-mêmes du contexte. Donc il nous faut généraliser presque toutes les catégories en **catégories relatives à un contexte**. En même temps, nous pouvons laisser les domaines dépendre du contexte. Nous aurons, par exemple,

$\mathrm{P}(\Gamma)$	où	Γ : cont	interprétée	prop/Γ
$\mathrm{SN}(\Gamma, A)$	où	Γ : cont, A : set/Γ	interprétée	$((A)\mathrm{prop})\mathrm{prop}/\Gamma$
$\mathrm{V1}(\Gamma, A)$	où	Γ : cont, A : set/Γ	interprétée	$(A)\mathrm{prop}/\Gamma$

Les règles grammaticales sont généralisées en conséquence. La règle de prédication devient

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{set}/\Gamma \quad Q : \mathrm{SN}(\Gamma, A) \quad F : \mathrm{V1}(\Gamma, A)}{\begin{array}{l}\mathrm{PrédV1}(\Gamma, A, Q, F) : \mathrm{P}(\Gamma) \\ \mathrm{PrédV1}(\Gamma, A, Q, F)^\star = Q^\star(F^\star) \\ \mathrm{PrédV1}(\Gamma, A, Q, F)^\circ(m) = Q^\circ(-)\ F^\circ(\mathrm{gen}(Q), \mathrm{num}(Q), m)\end{array}}$$

Dans cette règle, comme dans la plupart des règles, le contexte est le même pour tous les constituants. L'usage essentiel des contextes est montré par les règles qui permettent l'extension du contexte. Par exemple, la règle pour la formation d'une implication permet l'extension du contexte par une preuve de l'antécédent. Le conséquent est formé dans le contexte étendu :

$\mathrm{P} \rightarrow \mathit{si}\ \mathrm{P},\ \mathrm{P}$

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{P}(\Gamma) \quad B : \mathrm{P}((\Gamma,\ x : A^\star))}{\begin{array}{l}\mathrm{si}(\Gamma, A, B) : \mathrm{P}(\Gamma) \\ \mathrm{si}(\Gamma, A, B)^\star = (\Pi x : A^\star)B^\star \\ \mathrm{si}(\Gamma, A, B)^\circ(m) = \mathit{si}(A^\circ(\mathrm{ind})),\ B^\circ(m)\end{array}}$$

Dans la règle d'interprétation, l'opérateur Π lie la nouvelle variable x qui peut être libre dans $B^\star$, pour former une proposition dans le contexte originel Γ.

Il y a une analogie entre le paramètre de contexte de nos catégories syntaxiques et le paramètre du **monde possible** chez Montague, dans l'interprétation des arbres. Puisque ce paramètre n'est pas explicite dans la syntaxe de Montague, il ne peut pas y avoir de règles qui chez lui étendent le contexte.

1.7 *Les arguments syntaxiques et les arguments sémantiques*

Répétons la forme générale d'une règle grammaticale, avec toutes les additions faites depuis la section 1.3 — les paramètres de linéarisation $p_1, \ldots, p_m$ et les traits inhérents $T_1, \ldots, T_k$:

$$\frac{a_1 : \alpha_1 \cdots a_n : \alpha_n}{\begin{array}{l}C(a_1, \ldots, a_n) : \alpha \\ C(a_1, \ldots, a_n)^\star = F(a_1^\star, \ldots, a_n^\star) \\ C(a_1, \ldots, a_n)^\circ(p_1, \ldots, p_m) = G(T_{11}(a_1), \ldots, T_{nk_n}(a_n), p_1, \ldots, p_m, a_1^\circ, \ldots, a_n^\circ) \\ T_1(C(a_1, \ldots, a_n)) = t_1 \\ \ldots \\ T_k(C(a_1, \ldots, a_n)) = t_k\end{array}}$$

Ainsi la règle doit définir les traits inhérents — ceux qui appartiennent à la catégorie α — pour l'arbre formé. La linéarisation de l'arbre peut dépendre des traits inhérents des constituants $a_1, \ldots, a_n$, ainsi que des paramètres de la catégorie α.

Dans la règle, quelques-unes des catégories $\alpha_1, \ldots, \alpha_n$ sont des catégories grammaticales, telles que P, SN, tandis que d'autres sont des types ordinaires, telles que cont, set. Seules les catégories grammaticales sont représentées dans les règles syntagmatiques. Donc, si $\alpha_{i_1}, \ldots, \alpha_{i_p}$ sont les catégories grammaticales de la règle ci-dessus, elle correspond à la règle syntagmatique

$$\alpha \rightarrow \alpha_{i_1} \ \ldots \ \alpha_{i_p}.$$

Les constituants correspondants, $a_{i_1}, \ldots, a_{i_p}$ seront appelés les **arguments syntaxiques** de la règle et, ainsi, de l'opérateur C. Les autres constituants sont ses **arguments sémantiques**. Donc les règles syntagmatiques ne montrent que les arguments syntaxiques des constructions grammaticales, et en suppriment l'information sémantique.

Cette distinction nous donne un principe concernant les règles de linéarisation :

> La linéarisation supprime tous les arguments sémantiques, et eux seuls.

Que les arguments sémantiques doivent être supprimés est clair : la linéarisation n'est définie que pour les arguments syntaxiques. On sait comment linéariser une phrase, ou un verbe, mais on ne linéarise pas un contexte ou un ensemble. Dans l'autre direction, ce principe n'est nécessaire pour aucune raison formelle, mais il signifie que dans la construction grammaticale, les constituants sont toujours combinés, jamais effacés. Cela est précisément comme dans la grammaire syntagmatique. Donc, s'il y a de l'information cachée, ou supprimée - telle que l'interprétation d'une expression anaphorique — elle doit être exprimée par un argument sémantique. Cette conséquence n'est pas, en effet, une restriction à l'expressivité, mais au contraire : une prémisse syntaxique $a : \alpha$ est plus forte que la prémisse sémantique $a : \alpha^\star$, parce que $a : \alpha$ implique $a^\star : \alpha^\star$, mais il n'y a aucune opération inverse. Si on remplace une prémisse syntaxique par une prémisse sémantique, on obtient une règle plus forte.[31]

Un autre principe auquel nous obéirons est la **compositionnalité**, et de l'interprétation et de la linéarisation. Il dit que les équations qui les définissent ne dépendent pas des formes des arguments syntaxiques. En d'autres termes :

> L'interprétation d'un arbre est retenue dans l'interprétation de toute construction dont l'arbre est un constituant.
>
> La linéarisation d'un arbre est retenue dans la linéarisation de toute construction dont l'arbre est un constituant.

Si une forme d'arbres n'admet pas d'interprétation ou de linéarisation compositionnelles, elle n'est pas une structure réelle, mais peut-être une confusion de plusieurs structures, qui doit être analysée plus profondement.

[31] Pour être précis, on n'obtient pas toujours une règle plus forte, parce que les traits morphologiques ne sont pas définis pour les objets sémantiques. Par exemple, à partir d'un nom commun on obtient un genre grammatical, mais on n'en obtient pas à partir d'un ensemble. Voici un problème qui se posera dans la section 3.4.

1.8 *Les catégories grammaticales*

Le système des catégories syntaxiques doit faire des distinctions sémantiques qui ne sont pas habituelles dans la grammaire traditionnelle, ainsi que des distinctions syntaxiques que la logique ne connaît pas.

Les distinctions sémantiques sont assez connues dans la tradition de la formalisation logique. Ainsi Frege a observé, dans la *Begriffsschrift*, que les sujets des deux phrases

> *le nombre* 20 *peut être représenté comme la somme de quatre carrés,*
>
> *tout nombre entier positif peut être représenté comme la somme de quatre carrés,*

sont de types différents (Frege 1879, § 9). L'expression *le nombre* 20 est, logiquement, du type N, tandis que *tout nombre entier positif* est du type $((N)\text{prop})\text{prop}$. Il faut donc faire une distinction, dans la grammaire, entre les noms propres et les syntagmes nominaux formés par quantification.

Une autre distinction qu'il faut faire est celle entre les différents usages des noms communs. Un nom commun correspond parfois à un domaine d'individus, parfois à une fonction dont la valeur est un domaine, parfois à une fonction dont la valeur est un individu :

> *tout* **nombre** *est pair ou impair,*
>
> *les* **racines** *de cette équation sont complexes,*
>
> *le* **carré** *de* 9 *est* 81,
>
> *la* **somme** *de* 2 *et de* 3 *est* 5.

Il n'y a pas, en logique, de catégorie qui corresponde uniformément à la catégorie des noms communs, mais plusieurs catégories distinctes.

Dans notre grammaire, les distinctions de domaine et de contexte sont des distinctions sémantiques ultérieures, qui ne sont pas faites dans les grammaires basées sur la théorie simple des types. Ces distinctions introduisent une infinité de catégories, puisque toute instance d'une famille de catégories est une catégorie.

Les distinctions syntaxiques ont été introduites par Montague sous le titre **dissociation syntaxique** (*syntactic splitting*; v. Montague 1974, p. 249). Il la considère comme essentielle pour une grammaire d'une langue naturelle : un système de catégories purement logique, comme celui d'Ajdukiewicz (1935), ne suffit pas pour la syntaxe. Le système peut être différent pour les langues différentes.[32]

Dans notre grammaire, un exemple de la dissociation syntaxique est la distinction entre les verbes intransitifs et les adjectifs à une place. Tous les deux correspondent au même type sémantique, aux prédicats à une place, mais les verbes ne sont pas utilisés dans les combinaisons syntaxiques de la même façon que les adjectifs (cf. section 2.2). Un autre exemple est la distinction entre les noms propres, les termes formels qui désignent des individus, et les variables explicites : les variables peuvent être utilisées comme termes, et les termes comme noms, mais non inversement (cf. section 3.4).

[32]Montague ne mentionne pas la dissociation syntaxique antérieure à la sienne, effectuée par Bar-Hillel (1951) et développée par Lambek (1958). Dans leur système, la seule dissociation est la division du type des fonctions $(\alpha)\beta$ en deux, la catégorie β/α des préfixes et la catégorie $\alpha\backslash\beta$ des postfixes.

2 LE FRANÇAIS EN GÉNÉRAL

2.1 *Les opérations morphologiques*

Dans les règles de linéarisation, nous nous servons d'opérations qui prennent, comme arguments, des chaînes et des traits morphologiques, pour en produire des chaînes. Ces **opérations morphologiques** incluent les conjugaisons des verbes et les déclinaisons des noms, des adjectifs, et des pronoms. Dans cette section, nous présentons celles qui sont particulières au français ; quelques opérations générales sur les chaînes ont été définies dans l'introduction.

Une opération particulière est la méthode qui décide si une chaîne est **vocalique**, c.-à-d. si elle commence par une voyelle. À cause du *h* muet, cette opération n'est ni générale ni une fonction simple de l'orthographe conventionnelle. Nous la présupposons ici, sans en donner de définition détaillée.[33]

S'il est possible de décider si une chaîne est vocalique, on peut définir les mots qui subissent l'élision devant les chaînes vocaliques comme des opérations sur chaînes :

$$de, ne, que, se : \text{(chaîne)chaîne.}$$

Par exemple, *de*(*impair*) = *d'impair*, *de*(*point*) = *de point.*

L'opération

$$si : \text{(chaîne)chaîne}$$

produit une chaîne introduite par *s'* ou par *si* , selon que ce qui suit commence par le pronom *il, ils* ou par quelque autre mot.[34]

Une **conjugaison** est une fonction d'une chaîne et des paramètres du verbe, par exemple,

$$\text{verbe}_{22} : \text{(genre)(nombre)(mode)(chaîne)chaîne,}$$

qui est employée pour produire les formes des verbes tels que *contenir*: *contient, contiennent, contienne, contiennent.* Nous suivons la numérotation du Petit Robert, sauf pour le verbe *être*, qui est représenté comme une opération pour soi,

$$\textit{être} : \text{(nombre)(mode)chaîne.}$$

Les quatre formes utilisées pour *être* sont *est, sont, soit, soient.*

Les **déclinaisons** des noms et des adjectifs sont représentées d'une façon analogue, par exemple,

$$\text{nomrég} : \text{(nombre)(chaîne)chaîne,}$$

$$\text{adj}_{al} : \text{(genre)(nombre)(chaîne)chaîne.}$$

---

[33]Dans la représentation formelle des chaînes, on peut tout simplement introduire un caractère spécial pour le *h* muet, qui s'imprime justement comme *h*.

[34]La forme *s'* n'est pas produite précisément par la suite de caractères *il* ou *ils*, mais seulement par les pronoms masculins de ces formes. Une définition précise de *si* devrait donc utiliser l'induction sur tous les arbres représentant une phrase, parce que les chaînes ne sont reconnues comme des pronoms que par l'analyse syntaxique. Mais il y a une solution beaucoup moins coûteuse utilisable dans l'engendrement : dans les règles de linéarisation des pronoms, on se sert d'un caractère spécial qui s'imprime justement comme *i*. Dans les règles qui introduisent la conjonction *si*, ce caractère est reconnu comme produisant l'élision du *i*.

L'opération nomrég produit les formes *point, points* de la chaîne *point.* L'opération adj_{al} produit les formes *égal, égale, égaux, égales* de la chaîne *égal.* Puisqu'il n'y a pas de numérotation établie analogue à celle des conjugaisons, nous désignons les déclinaisons par les formes typiques ou par l'étiquette rég, pour «régulier».

Les pronoms et les autres mots structuraux sont représentés comme des fonctions des paramètres dont leurs formes dépendent: par exemple,

un, aucun : (genre)chaîne,

lui, il, lui-même, quel, tel, tout : (genre)(nombre)chaîne,

lequel : (préfixe)(genre)(nombre)chaîne.

Par exemple, *un* produit les deux formes *un, une*; *tel* produit les quatre formes *tel, telle, tels, telles*; *lequel* produit les formes *lequel, de laquelle, auxquelles* et beaucoup d'autres.

L'article défini prend, outre les paramètres, la chaîne suivante comme argument, parce que sa forme dépend de la vocalicité de celle-ci.

le : (préfixe)(genre)(nombre)(chaîne)chaîne.

Cette opération retourne aussi la chaîne à laquelle elle est appliquée:

$le(\grave{a}, \text{masc}, \text{sg}, point) = au\ point$, $le(\grave{a}, \text{masc}, \text{sg}, ordre) = \grave{a}\ l'ordre$,
$le(-, \text{masc}, \text{pl}, points) = les\ points$, $le(\grave{a}, \text{masc}, \text{pl}, points) = aux\ points$.

L'article indéfini ne dépend pas de la chaîne, mais il dépend du préfixe, parce qu'au pluriel, la préposition *de* l'efface:

indéf : (préfixe)(genre)(nombre)chaîne.

Par exemple,

$\text{indéf}(-, \text{fém}, \text{sg}) = une$, $\text{indéf}(\grave{a}, \text{fém}, \text{sg}) = \grave{a}\ une$, $\text{indéf}(de, \text{fém}, \text{sg}) = d'une$,
$\text{indéf}(-, \text{fém}, \text{pl}) = des$, $\text{indéf}(\grave{a}, \text{fém}, \text{pl}) = \grave{a}\ des$, $\text{indéf}(de, \text{fém}, \text{pl}) = de$.

Le **genre collectif** est une opération qui prend une liste de genres comme argument et retourne un genre, qui est le féminin si chaque élément du liste est le féminin, autrement le masculin,

$$\text{gencoll} : (\text{list}_1(\text{genre}))\text{genre}.$$

L'opération conjform, qui forme la conjonction d'une liste de chaînes a été définie dans l'introduction. Une application assez fréquente de conjform est celle où la conjonction est *et*,

$$\text{etform}(l) = \text{conjform}(et, l).$$

Une autre forme de conjonction d'une liste, assez spéciale, est

$$\text{etcoll} : (\text{préfixe})(\text{list}_2(\text{préfixe} \rightarrow \text{chaîne}))\text{chaîne},$$

qui distribue la plupart des prépositions à tous les éléments de la liste, mais ne distribue pas *entre.*

2.2 *Les catégories de prédication*

Comme premier groupe de catégories discutées en détail, nous prendrons les catégories qui correspondent aux propositions et aux fonctions propositionnelles: la **phrase**, les **verbes** et les **adjectifs**.

Un prédicat à une place peut être exprimé soit par un verbe soit par un adjectif:

la série harmonique diverge,

la série harmonique est divergente.

Pour un prédicat à deux places, il y a plusieurs possibilités:

le verbe transitif: *l coupe m*,

le verbe à un complément prépositionnel: *f tend vers y*,[35]

le verbe collectif: *l et m se croisent*,

l'adjectif à un complément prépositionnel: *n est supérieur à m*,

l'adjectif comparatif: *n est plus grand que m*,[36] [37]

l'adjectif collectif: *l et m sont parallèles.*[38] [39]

Outre la similarité sémantique, qu'il y a entre toutes les catégories interprétées comme le même type sémantique, il y a des rapports grammaticaux indépendants de l'interprétation. Ainsi tous les verbes se conjuguent en genre, nombre et mode, tous les adjectifs en genre et nombre[40]; les adjectifs sont appliqués par le moyen des **copules**, qui sont conjuguées en genre[41], nombre et mode; les adjectifs peuvent aussi modifier les noms communs.[42]

Une phrase à deux arguments nominaux peut être analysée de deux manières, selon que le sujet ou le complément est pensé comme l'élément ajouté en dernier:

---

[35]Cette catégorie contiendra les verbes transitifs comme cas spécial.

[36]Cette catégorie n'est pas un cas spécial de la précédente, puisque *que* ne fonctionne pas comme une préposition: on ne peut pas dire *nombre que lequel n est plus grand.*

[37]Notons qu'un adjectif comparatif n'est pas, en général, dérivé d'un adjectif à une place: il n'y a pas d'adjectif *grand* dont *plus grand* soit une forme, parce qu'il n'y a pas de prédicat mathématique correspondant. Au contraire, s'il y a un usage absolu de *grand*, il est logiquement dérivé du comparatif: par exemple, la phrase *x est assez grand pour que f(x) soit positif* est interprétée par rapport à une relation d'ordre.

[38]Les catégories collectives sont encore un produit des distinctions logiques. Si on ne voulait qu'engendrer des chaînes françaises, on n'aurait peut-être pas besoin de distinguer entre les formes des phrases

l et m sont verticales,

l et m sont parallèles,

dont la première est la conjonction de deux propositions, formée par la coordination, mais la seconde est formée par prédication collective.

[39]La prédication collective en français n'est pas toujours analysable par un prédicat à deux places: dans la phrase 2, 3 *et* 5 *sont les facteurs de* 30 il s'agit, plutôt, d'un prédicat à une place dont l'argument est une liste.

[40]Le singulier des verbes et des adjectifs collectifs n'est guère usité: nous omettons leur paramètre de nombre.

[41]Le genre des copules n'est pas exemplifié dans cet article, mais il est trouvé dans les formes composées de *devenir*.

[42]La modification sera discutée plus tard, ainsi que la question si la fonction primaire des adjectifs est la prédication ou la modification.

l (*coupe m*),

(*l coupe*) *m*.

La première de ces structures a déjà été trouvée, en analysant (*coupe m*) comme un verbe à une place. Pour trouver la seconde, il faut introduire une nouvelle catégorie des **phrases sans objets**, dont les expressions sont interprétées comme des prédicats à une place.[43]

Résumons les catégories de prédication dans un tableau. Les paramètres dans le tableau sont typés

$$\Gamma : \text{cont},\ A, B : \text{set}/\Gamma,\ g : \text{genre},\ n : \text{nombre},\ m : \text{mode},\ r : \text{préfixe}.$$

Aucune des catégories de ce tableau-ci n'a de traits inhérents, donc la dernière colonne est vide.

catégorie	symbole	interprétation	param.	inhér.
phrase	$\mathrm{P}(\Gamma)$	prop/Γ	m	
verbe à 1 place	$\mathrm{V1}(\Gamma, A)$	$(A)\mathrm{prop}/\Gamma$	g, n, m	
verbe à 2 places	$\mathrm{V2}(\Gamma, A, B, r)$	$(A)(B)\mathrm{prop}/\Gamma$	g, n, m	
verbe collectif	$\mathrm{VC}(\Gamma, A, B)$	$(A)(B)\mathrm{prop}/\Gamma$	g, m	
phrase sans objet	$\mathrm{V}^1/_2(\Gamma, B, r)$	$(B)\mathrm{prop}/\Gamma$	m	
copule	Cp	$(X : \mathrm{set})((X)\mathrm{prop})(X)\mathrm{prop}$	g, n	
adjectif à 1 place	$\mathrm{A1}(\Gamma, A)$	$(A)\mathrm{prop}/\Gamma$	g, n	
adj. à 2 places	$\mathrm{A2}(\Gamma, A, B, r)$	$(A)(B)\mathrm{prop}/\Gamma$	g, n	
adj. collectif	$\mathrm{AC}(\Gamma, A, B)$	$(A)(B)\mathrm{prop}/\Gamma$	g	
adj. comparatif	$\mathrm{Aq}(\Gamma, A, B)$	$(A)(B)\mathrm{prop}/\Gamma$	g, n	

2.3 *Les règles de prédication*

Une phrase complète construite avec les catégories que nous venons de définir a une des formes suivantes :

SN V1,

SN V2 SN,

SNC VC,

SN Cp A1,

SN Cp A2 SN,

SNC Cp AC,

SN Cp Aq *que* SN.

[43]Cette catégorie aura, au moins, trois usages : 1° Il est possible de coordonner des expressions de cette forme : *l coupe mais k ne coupe pas m*. 2° On peut former les propositions relatives telles que *que l coupe*. 3° Si les arguments sont des quantificateurs, leurs portées peuvent se relier l'une à l'autre de deux façons. Les usages 1° et 2° sont la motivation des catégories « slash » de la grammaire syntagmatique généralisée ; notre catégorie $\mathrm{V}^1/_2$ correspondra aux catégories P/SN et P/SP de cette théorie (cf. Gazdar *et al.* 1985, chapitre 7, et Abeillé 1993, section 2.5). Le troisième phénomène, connu comme « scope ambiguity » des quantificateurs, est expliqué par une technique très différente et, à notre avis, trop générale, de liaison des variables dans Montague (1974), tandis que les possibilités offertes par le calcul de Lambek permettent une analyse analogue à la nôtre, bien que plus générale (cf. Morrill 1994, p. 84).

La catégorie SN des syntagmes nominaux à déjà été définie ; la catégorie $\mathrm{SNC}(\Gamma, A, B)$ des **syntagmes nominaux collectifs** est analogue, interprétée $((A)(B)\mathrm{prop})\mathrm{prop}/\Gamma$.[44]

On pourrait stipuler une règle de prédication pour chacune de ces formes, mais il est plus efficace de formuler des règles qui construisent les arbres par des pas minimaux, en n'ajoutant qu'un élément à la fois.[45] Pour les verbes à une place, cet élément est le sujet :

$\mathrm{P} \rightarrow \mathrm{SN}\ \mathrm{V1}$

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{set}/\Gamma \quad Q : \mathrm{SN}(\Gamma, A) \quad F : \mathrm{V1}(\Gamma, A)}{\begin{array}{l}\mathrm{PrédV1}(\Gamma, A, Q, F) : \mathrm{P}(\Gamma) \\ \mathrm{PrédV1}(\Gamma, A, Q, F)^\star = Q^\star(F^\star) \\ \mathrm{PrédV1}(\Gamma, A, Q, F)^\circ(m) = Q^\circ(-)\ F^\circ(\mathrm{gen}(Q), \mathrm{num}(Q), m)\end{array}}$$

À un verbe à deux places, on peut ajouter ou le complément, pour former un verbe à une place, ou le sujet, pour former une phrase sans objet.

$\mathrm{V1} \rightarrow \mathrm{V2}\ \mathrm{SN}$

$$\frac{\Gamma : \mathrm{cont} \quad A, B : \mathrm{set}/\Gamma \quad r : \text{préfixe} \quad F : \mathrm{V2}(\Gamma, A, B, r) \quad Q : \mathrm{SN}(\Gamma, B)}{\begin{array}{l}\mathrm{ComplV2}(\Gamma, A, B, r, F, Q) : \mathrm{V1}(\Gamma, A) \\ \mathrm{ComplV2}(\Gamma, A, B, r, F, Q)^\star = (x)Q^\star((y)F^\star(x, y)) \\ \mathrm{ComplV2}(\Gamma, A, B, r, F, Q)^\circ(g, n, m) = F^\circ(g, n, m)\ Q^\circ(r)\end{array}}$$

$\mathrm{V}^1/_2 \rightarrow \mathrm{SN}\ \mathrm{V2}$

$$\frac{\Gamma : \mathrm{cont} \quad A, B : \mathrm{set}/\Gamma \quad r : \text{préfixe} \quad Q : \mathrm{SN}(\Gamma, A) \quad F : \mathrm{V2}(\Gamma, A, B, r)}{\begin{array}{l}\mathrm{FractV2}(\Gamma, A, B, r, Q, F) : \mathrm{V}^1/_2(\Gamma, B, r) \\ \mathrm{FractV2}(\Gamma, A, B, r, Q, F)^\star = (y)Q^\star((x)F^\star(x, y)) \\ \mathrm{FractV2}(\Gamma, A, B, r, Q, F)^\circ(m) = Q^\circ(-)\ F^\circ(\mathrm{gen}(Q), \mathrm{num}(Q), m)\end{array}}$$

Une phrase sans objet est complétée en une phrase en y ajoutant l'objet :

$\mathrm{P} \rightarrow \mathrm{V}^1/_2\ \mathrm{SN}$

$$\frac{\Gamma : \mathrm{cont} \quad B : \mathrm{set}/\Gamma \quad r : \text{préfixe} \quad F : \mathrm{V}^1/_2(\Gamma, B, r) \quad Q : \mathrm{SN}(\Gamma, B)}{\begin{array}{l}\mathrm{ComplV}^1/_2(\Gamma, B, r, F, Q) : \mathrm{P}(\Gamma) \\ \mathrm{ComplV}^1/_2(\Gamma, B, r, F, Q)^\star = Q^\star(F^\star) \\ \mathrm{ComplV}^1/_2(\Gamma, B, r, F, Q)^\circ(m) = F^\circ(m)\ Q^\circ(r)\end{array}}$$

Un verbe collectif ne peut être complété que par un sujet qui est un syntagme nominal collectif.

$\mathrm{P} \rightarrow \mathrm{SNC}\ \mathrm{VC}$

$$\frac{\Gamma : \mathrm{cont} \quad A, B : \mathrm{set}/\Gamma \quad Q : \mathrm{SNC}(\Gamma, A, B) \quad F : \mathrm{VC}(\Gamma, A, B)}{\begin{array}{l}\mathrm{PrédVC}(\Gamma, A, B, Q, F) : \mathrm{P}(\Gamma) \\ \mathrm{PrédVC}(\Gamma, A, B, Q, F)^\star = Q^\star(F^\star) \\ \mathrm{PrédVC}(\Gamma, A, B, Q, F)^\circ(m) = Q^\circ(-)\ F^\circ(\mathrm{gen}(Q), m)\end{array}}$$

[44] Les catégories nominales seront discutées en détail dans la section 2.7.

[45] V. section 4.4 pour une définition des cadres de prédication, c.-à-d. des macros qui appliquent les prédicats à tous leurs arguments à la fois.

Un adjectif peut être complété en un verbe de la catégorie correspondante en y ajoutant une copule. Les traits verbaux de ce syntagme verbal sont hérités et par la copule et par l'adjectif, sauf le mode, qui n'est défini que pour la copule.

V1 $\rightarrow$ Cp A1

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad C : \text{Cp} \quad F : \text{A1}(\Gamma, A)}{\begin{array}{l}\text{PrédA1}(\Gamma, A, C, F) : \text{V1}(\Gamma, A) \\ \text{PrédA1}(\Gamma, A, C, F)^\star = C^\star(F^\star) \\ \text{PrédA1}(\Gamma, A, C, F)^\circ(g, n, m) = C^\circ(g, n, m)\ F^\circ(g, n)\end{array}}$$

V2 $\rightarrow$ Cp A2

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad r : \text{préfixe} \quad C : \text{Cp} \quad F : \text{A2}(\Gamma, A, B, r)}{\begin{array}{l}\text{PrédA2}(\Gamma, A, B, r, C, F) : \text{V2}(\Gamma, A, B, r) \\ \text{PrédA2}(\Gamma, A, B, r, C, F)^\star = (x)(y)C^\star((z)F^\star(z, y), x) \\ \text{PrédA2}(\Gamma, A, B, r, C, F)^\circ(g, n, m) = C^\circ(g, n, m)\ F^\circ(g, n)\end{array}}$$

VC $\rightarrow$ Cp AC

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad C : \text{Cp} \quad F : \text{AC}(\Gamma, A, B)}{\begin{array}{l}\text{PrédAC}(\Gamma, A, B, C, F) : \text{VC}(\Gamma, A, B) \\ \text{PrédAC}(\Gamma, A, B, C, F)^\star = (x)(y)C^\star((z)F^\star(z, y), x) \\ \text{PrédAC}(\Gamma, A, B, r, C, F)^\circ(g, m) = C^\circ(g, \text{pl}, m)\ F^\circ(g)\end{array}}$$

Cette manière de combinaison n'est pas définie pour l'adjectif comparatif, qui n'a pas de catégorie verbale correspondante. On peut seulement y ajouter un complément introduit par *que*, pour former un adjectif à une place. La même possibilité existe pour A2, aussi.

A1 $\rightarrow$ Aq *que* SN

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad F : \text{Aq}(\Gamma, A, B) \quad Q : \text{SN}(\Gamma, B)}{\begin{array}{l}\text{ComplAq}(\Gamma, A, B, F, Q) : \text{A1}(\Gamma, A) \\ \text{ComplAq}(\Gamma, A, B, F, Q)^\star = (x)Q^\star((y)F^\star(x, y)) \\ \text{ComplAq}(\Gamma, A, B, F, Q)^\circ(g, n) = F^\circ(g, n)\ que(Q^\circ(-))\end{array}}$$

A1 $\rightarrow$ A2 SN

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad r : \text{préfixe} \quad F : \text{A2}(\Gamma, A, B, r) \quad Q : \text{SN}(\Gamma, B)}{\begin{array}{l}\text{ComplA2}(\Gamma, A, B, r, F, Q) : \text{A1}(\Gamma, A) \\ \text{ComplA2}(\Gamma, A, B, r, F, Q)^\star = (x)Q^\star((y)F^\star(x, y)) \\ \text{ComplA2}(\Gamma, A, B, r, F, Q)^\circ(g, n) = F^\circ(g, n)\ Q^\circ(r)\end{array}}$$

Les règles citées comprennent toutes les combinaisons de deux éléments adjacents.[46] Puisqu'il y a des ordres alternatifs dans lesquels les combinaisons peuvent être effectuées, les phrases peuvent avoir des analyses alternatives. Ainsi l'énoncé

[46] Avec l'exception de la combinaison sujet+copule, pour laquelle il nous faudrait une nouvelle catégorie, analogue à $V^1/2$.

6 *est supérieur à* 4

résulte de trois arbres distincts,[47]

PrédV1(6, PrédA1(être(vrai), ComplA2(*à*, supérieur, 4))),
PrédV1(6, ComplV2(PrédA2(*à*, être(vrai), supérieur), 4)),
ComplV$^1/_2$(FractV2(6, PrédA2(*à*, être(vrai), supérieur)), 4).

Tous ces arbres ont la même interprétation,

$$6 > 4,$$

mais il y a aussi des chaînes où l'ordre de combinaison est signifiant. Par exemple,

3 *et* 6 *sont supérieurs à* 2 *ou à* 4

a les deux interprétations

$(3 > 2 \vee 3 > 4)\ \&\ (6 > 2 \vee 6 > 4)$,
$(3 > 2\ \&\ 6 > 2) \vee (3 > 4\ \&\ 6 > 4)$,

selon l'ordre de combinaison.[48]

2.4 *Sur la nature des adjectifs*

L'adjectif à une place a deux fonctions principales : la prédication, comme dans la phrase

3 *est impair,*

et la **modification**, comme dans le nom commun complexe

nombre impair.

Ces deux fonctions peuvent être traitées sans difficulté par l'interprétation des expressions de A1(A) comme des prédicats sur A,

(A)prop

(nous pouvons ignorer le contexte pour le moment). La modification sera définie dans la section 2.10.

Notre interprétation des adjectifs correspond directement à leur fonction prédicative. Ce qui correspondrait directement à leur fonction modificative serait le type

(set)set,

[47]Les arguments sémantiques des arbres sont supprimés dans la plupart de nos exemples. Ici, nous supposons l'adjectif *supérieur* de la catégorie A2(( ), N, N, *à*) déjà discuté dans la section 1.5 et la copule *être*, définie dans la section 2.5. Nous surchargeons les chiffres en les utilisant et comme chaînes, et comme arbres, et comme éléments de N, s'il n'y a pas de risque de confusion.

[48]Une ambiguïté syntaxique sans ambiguïté sémantique est connue comme la fausse ambiguïté, parfois considérée comme une propriétée indésirable d'une grammaire. Dans le calcul de Lambek, elle est très fréquente, parce que n'importe quelle paire de mots adjacents peut être considérée comme constituant. Pourtant, il n'est pas étonnant qu'une distinction structurelle opérante dans certaines situations soit neutralisée dans d'autres. Cf. Lecomte (1994, pp. 2–5).

puisque les noms communs sont interprétés comme des ensembles. Cette interprétation des adjectifs est standard dans la grammaire catégorielle ; v. par exemple Morrill (1994, p. 146). Mais ce type ne fait pas de référence à un domaine : donc il ne peut pas exprimer les restrictions de sélection des adjectifs. Un adjectif pourrait modifier un nom quelconque, de sorte qu'on pourrait dire *droite impaire*, *nombre vertical* — des énoncés sans interprétation mathématique. La même liberté concernerait la prédication : on pourrait dire 3 *est vertical*, etc.

Nous concluons que c'est l'interprétation prédicative qu'il faut choisir, pour qu'une analyse exacte des adjectifs soit possible. Cette solution rend les adjectifs sémantiquement analogues aux verbes, mais nous avons déjà constaté une analogie syntaxique assez forte : le parallélisme entre les catégories V1, V2 et VC, d'une part, et A1, A2 et AC, de l'autre.[49]

2.5 *La négation*

La négation du verbe, en français, est un peu particulière — non seulement parce qu'elle est exprimée par deux éléments disjoints, *ne* et *pas*, mais aussi parce qu'elle ne s'applique qu'aux verbes primitifs, ou lexicaux : on ne peut pas mettre un syntagme verbal complexe entre les deux élément de la négation, disant

l ne coupe m pas,

3 *n'est pair pas,*

la série ne diverge et converge pas.

Ainsi il faut reconnaître des **catégories lexicales** de verbes,

catégorie	symbole	interprétation	param.	inhér.
verbe à 1 place	$\mathrm{V1L}(A)$	$(A)\mathrm{prop}$	g, n, m	
verbe à 2 places	$\mathrm{V2L}(A, B, r)$	$(A)(B)\mathrm{prop}$	g, n, m	
verbe collectif	$\mathrm{VCL}(A, B)$	$(A)(B)\mathrm{prop}$	g, m	

Le dépendance du contexte n'est pas nécessaire, puisqu'un élément lexical ne peut pas contenir de constituants qui produisent des dépendances.[50]

La négation est, ainsi, toujours formée d'un verbe lexical, par exemple,

$\mathrm{V1} \rightarrow$ *ne* $\mathrm{V1L}$ *pas*

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{set} \quad F : \mathrm{V1L}(A)}{\begin{array}{l}\mathrm{NégV1}(\Gamma, A, F) : \mathrm{V1}(\Gamma, A) \\ \mathrm{NégV1}(\Gamma, A, F)^{\star} = (x) \sim F^{\star}(x) \\ \mathrm{NégV1}(\Gamma, A, F)^{\circ}(g, n, m) = ne(F^{\circ}(g, n, m))\ pas\end{array}}$$

[49]Dans la théorie simple des types, les noms communs sont interprétés comme des fonctions propositionnelles, $(D)\mathrm{prop}$, et il est facile de voir comment n'importe laquelle des deux fonctions de l'adjectif pourrait être traitée comme définitionnelle. Dans la théorie constructive des types, on pourrait suivre cette idée, en rendant les noms communs dépendants du domaine. La catégorie $\mathrm{CN}(A)$ serait interprétée comme $(A)\mathrm{prop}$, et ainsi, on pourrait interpréter $\mathrm{A1}(A)$ comme $((A)\mathrm{prop})(A)\mathrm{prop}$, sans perdre les restrictions de sélection. Mais cela ne serait pas naturel comme interprétation des noms communs qui servent comme des termes indicatifs du domaine de quantification et non comme des prédicats.

[50]Montague (1974, p. 250) fait une distinction, pour chaque catégorie, entre les « basic expressions » et les expressions en général. Une telle distinction n'est guère nécessaire dans la définition de l'anglais, mais elle peut être utile quand on définit des opérations sur la langue par récurrence, pour réduire le nombre des cas qu'il faut distinguer.

Puisque le verbe lexical a aussi un usage positif, il nous faut une règle qui donne un V1L à partir d'un V1 sans y ajouter rien.[51] Mais nous pouvons économiser en introduisant un paramètre de **polarité**, qui est une valeur booléenne, le vrai ou le faux :

V1 $\rightarrow$ (*ne*) V1L (*pas*)

$$\frac{\Gamma : \text{cont} \quad A : \text{set} \quad F : \text{V1L}(A) \quad v : \text{polarité}}{\begin{array}{l}\text{UseV1L}(\Gamma, A, F, v) : \text{V1}(\Gamma, A) \\ \text{UseV1L}(\Gamma, A, F, \text{vrai})^\star = F^\star \\ \text{UseV1L}(\Gamma, A, F, \text{faux})^\star = (x) \sim F^\star(x) \\ \text{UseV1L}(\Gamma, A, F, \text{vrai})^\circ(g, n, m) = F^\circ(g, n, m) \\ \text{UseV1L}(\Gamma, A, F, \text{faux})^\circ(g, n, m) = ne(F^\circ(g, n, m))\ pas\end{array}}$$

NégV1 est ainsi définissable comme la macro $(\Gamma)(A)(F)\text{UseV1L}(\Gamma, A, F, \text{faux})$.[52]

Les usages des verbes de V2L et de VCL, ainsi que de la copule *être*, sont définis de la même façon.

V2 $\rightarrow$ (*ne*) V2L (*pas*)

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set} \quad r : \text{préfixe} \quad F : \text{V2L}(A, B, r) \quad v : \text{polarité}}{\begin{array}{l}\text{UseV2L}(\Gamma, A, B, r, F, v) : \text{V2}(\Gamma, A, B, r) \\ \text{UseV2L}(\Gamma, A, B, r, F, \text{vrai})^\star = F^\star \\ \text{UseV2L}(\Gamma, A, B, r, F, \text{faux})^\star = (x)(y) \sim F^\star(x, y) \\ \text{UseV2L}(\Gamma, A, B, r, F, \text{vrai})^\circ(g, n, m) = F^\circ(g, n, m) \\ \text{UseV2L}(\Gamma, A, B, r, F, \text{faux})^\circ(g, n, m) = ne(F^\circ(g, n, m))\ pas\end{array}}$$

VC $\rightarrow$ (*ne*) VCL (*pas*)

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set} \quad F : \text{VCL}(A, B) \quad v : \text{polarité}}{\begin{array}{l}\text{UseVCL}(\Gamma, A, B, F, v) : \text{VC}(\Gamma, A, B) \\ \text{UseVCL}(\Gamma, A, B, F, \text{vrai})^\star = F^\star \\ \text{UseVCL}(\Gamma, A, B, F, \text{faux})^\star = (x)(y) \sim F^\star(x, y) \\ \text{UseVCL}(\Gamma, A, B, F, \text{vrai})^\circ(g, m) = F^\circ(g, m) \\ \text{UseVCL}(\Gamma, A, B, F, \text{faux})^\circ(g, m) = ne(F^\circ(g, m))\ pas\end{array}}$$

Cp $\rightarrow$ (*n'*)*est* (*pas*)

$$\frac{v : \text{polarité}}{\begin{array}{l}\text{être}(v) : \text{Cp} \\ \text{être}(\text{vrai})^\star = (Y)Y \\ \text{être}(\text{faux})^\star = (Y)(x) \sim Y(x) \\ \text{être}(\text{vrai})^\circ(g, n, m) = \hat{e}tre(n, m) \\ \text{être}(\text{faux})^\circ(g, n, m) = ne(\hat{e}tre(n, m))\ pas\end{array}}$$

[51] C'est ce que fait la règle S1 de Montague (1974, p. 251). Notons que sa règle de négation, dans S17 (p. 252), s'applique à tous les syntagmes verbaux, ce qui peut être justifié en anglais.

[52] La compositionnalité n'est pas violée, parce que l'argument v : polarité, dont nous distinguons des cas, n'est pas un argument syntaxique.

2.6 *Les formes réfléchies et réciproques*

Le complément d'un verbe peut être le pronom *lui-même*, dans une forme qui dépend des paramètres de genre et de nombre reçus par le verbe. Il fonctionne comme un **pronom réfléchi**,[53] qui récupère la référence du sujet. Ainsi le type du second argument du verbe doit être le même que celui du premier.

V1 $\rightarrow$ V2 *lui-même*

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad r : \text{préfixe} \quad F : \text{V2}(\Gamma, A, A, r)}{\begin{array}{l}\text{RéflV2}(\Gamma, A, r, F) : \text{V1}(\Gamma, A) \\ \text{RéflV2}(\Gamma, A, r, F)^{\star} = (x)F^{\star}(x, x) \\ \text{RéflV2}(\Gamma, A, r, F)^{\circ}(g, n, m) = F^{\circ}(g, n, m)\ r\ \textit{lui-même}(g, n)\end{array}}$$

Cette règle comprend, aussi, quelques usages réfléchis des adjectifs à deux places — à savoir, ceux formés par la règle PrédA2, par exemple,

2 n'est pas supérieur à lui-même

qui résulte de l'arbre

$$\text{PrédV1}(2, \text{RéflV2}(\grave{a}, \text{PrédA2}(\text{être}(\text{faux}), \text{supérieur}))).$$

Mais si l'adjectif est coordonné avec d'autres adjectifs, comme dans la phrase

2 n'est pas impair ou supérieur à lui-même

il faut utiliser une règle spéciale, avec l'effet

A1 $\rightarrow$ A2 *lui-même*,

dont la définition est analogue à RéflV2.

Pour les verbes transitifs, c.-à-d. pour les verbes de la catégorie V2(−), le pronom réfléchi atone, *se*, est aussi applicable. Pour que la négation devienne correcte, il faut restreindre cette structure aux verbes lexicaux, avec le paramètre de polarité déjà familier.[54]

V1 $\rightarrow$ (*ne*) *se* V2L(−) (*pas*)

$$\frac{\Gamma : \text{cont} \quad A : \text{set} \quad F : \text{V2L}(A, A, -) \quad v : \text{polarité}}{\begin{array}{l}\text{UseRéflV2L}(\Gamma, A, F, v) : \text{V1}(\Gamma, A) \\ \text{UseRéflV2L}(\Gamma, A, F, \text{vrai})^{\star} = (x)F^{\star}(x, x) \\ \text{UseRéflV2L}(\Gamma, A, F, \text{faux})^{\star} = (x) \sim F^{\star}(x, x) \\ \text{UseRéflV2L}(\Gamma, A, F, \text{vrai})^{\circ}(g, n, m) = \textit{se}(F^{\circ}(g, n, m)) \\ \text{UseRéflV2L}(\Gamma, A, F, \text{faux})^{\circ}(g, n, m) = \textit{ne}(\textit{se}(F^{\circ}(g, n, m)))\ \textit{pas}\end{array}}$$

Avec le **pronom réciproque**, *l'un l'autre*, les verbes de V2 peuvent être transformés en verbes collectifs. La forme atone *se* est applicable pour V2(−).

---

[53] Ici, « on attendrait logiquement » le pronom *soi-même* (Grevisse, § 640), mais le français moderne préfère le pronom personnel ordinaire.

[54] Nous omettons l'analyse plus avancée des pronoms atones et de leurs combinaisons.

VC $\rightarrow$ V2 *l'un l'autre*

Γ : cont A : set/Γ r : préfixe F : V2(Γ, A, A, r)

RéciprV2(Γ, A, r, F) : VC(Γ, A, A)
RéciprV2(Γ, A, r, F)* = $(x)(y)(F^\star(x,y)\&F^\star(y,x))$
RéciprV2(Γ, A, r, F)$^\circ(g,m)$ = $F^\circ(g, \text{pl}, m)$ *le*$(-, g, \text{sg}, un(g))$ *le*$(r, g, \text{sg}, autre)$

VC $\rightarrow$ (*ne*) *se* V2L($-$) (*pas*)

Γ : cont A : set F : V2L($A, A, -$) v : polarité

UseRéciprV2L(Γ, A, F, v) : VC(Γ, A, A)
UseRéciprV2L(Γ, A, F, vrai)* = $(x)(y)(F^\star(x,y)\&F^\star(y,x))$
UseRéciprV2L(Γ, A, F, faux)* = $(x)(y) \sim (F^\star(x,y)\&F^\star(y,x))$
UseRéciprV2L(Γ, A, F, vrai)$^\circ(g,m)$ = $se(F^\circ(g, \text{pl}, m))$
UseRéciprV2L(Γ, A, F, faux)$^\circ(g,m)$ = $ne(se(F^\circ(g, \text{pl}, m)))$ *pas*

Par exemple, les phrases

l et m se coupent,

l et m sont convergentes l'une avec l'autre,

expriment la même proposition,

$$\text{Con}(l,m)\&\text{Con}(m,l),$$

parce que le verbe *couper* et l'adjectif *convergent* ont la même interprétation, le prédicat Con.

2.7 *Le syntagme nominal*

Il y a plusieurs catégories nominales, c.-à-d. catégories dont les expressions sont utilisées dans la formation des **syntagmes nominaux**:

catégorie	symbole	interprétation	param.	inhér.
nom commun	CN(Γ)	set/Γ	n	gen
nom propre	PN(Γ, A)	A/Γ	r	gen,num
syntagme nominal	SN(Γ, A)	$((A)\text{prop})\text{prop}/\Gamma$	r	gen,num
déterminant	Dét	$(X : \text{set})((X)\text{prop})\text{prop}$	g, r	num
nom pr. collectif	PNC(Γ, A, B)	$A \times B/\Gamma$	r	gen
synt. nom. coll.	SNC(Γ, A, B)	$((A)(B)\text{prop})\text{prop}/\Gamma$	r	gen
fonction à 1 place	F1(Γ, A, D, r)	$(A)D/\Gamma$	n	gen
fonction à 2 pl.	F2(Γ, A, B, D, r, t)	$(A)(B)D/\Gamma$	n	gen
fonction collective	FC(Γ, A, B, D, r)	$(A)(B)D/\Gamma$	n	gen

Tous les noms propres que nous considérons auront le nombre singulier, mais on peut en trouver aussi des pluriels, par exemple, l'expression anaphorique *ces ciseaux*, synonyme de *cette paire de ciseaux*.[55] Les noms propres collectifs sont, pour ainsi dire, pluriels par nature — ou mieux, la question ne se pose pas.

[55]Notre concept du nom propre est celui du terme singulier — une expression qui désigne un individu et qui peut servir comme argument d'un verbe. Cette catégorie comprend aussi les démonstratifs qui désignent des individus dans un contexte.

Les formes les plus connues du syntagme nominal sont le nom propre **monté**[56] et le nom commun déterminé,

SN $\rightarrow$ PN

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad a : \text{PN}(\Gamma, A)}{\begin{array}{l}\text{Mont}(\Gamma, A, a) : \text{SN}(\Gamma, A) \\ \text{Mont}(\Gamma, A, a)^\star = (Y)Y(a^\star) \\ \text{Mont}(\Gamma, A, a)^\circ(r) = a^\circ(r) \\ \text{gen}(\text{Mont}(\Gamma, A, a)) = \text{gen}(a) \\ \text{num}(\text{Mont}(\Gamma, A, a)) = \text{num}(a)\end{array}}$$

SN $\rightarrow$ Dét CN

$$\frac{\Gamma : \text{cont} \quad D : \text{Dét} \quad A : \text{CN}(\Gamma)}{\begin{array}{l}\text{Déterm}(\Gamma, D, A) : \text{SN}(\Gamma, A^\star) \\ \text{Déterm}(\Gamma, D, A)^\star = D(A^\star) \\ \text{Déterm}(\Gamma, D, A)^\circ(r) = D^\circ(\text{gen}(A), r)\ A^\circ(\text{num}(D)) \\ \text{gen}(\text{Déterm}(\Gamma, D, A)) = \text{gen}(A) \\ \text{num}(\text{Déterm}(\Gamma, D, A)) = \text{num}(D)\end{array}}$$

Nous ne ferons pas beaucoup d'usage de la règle Déterm, parce que son interférence avec la négation est problématique. Le déterminant *quelque* se comporte de la manière attendue : l'énoncé

quelque nombre n'est pas pair

reçoit l'interprétation

$$(\exists x : N) \sim \text{Pair}(x).$$

Mais si *tout* est catégorisé comme un déterminant, l'énoncé

tout nombre n'est pas pair

reçoit l'interprétation

$$(\forall x : N) \sim \text{Pair}(x),$$

bien que son interprétation intuitive soit

$$\sim (\forall x : N) \text{Pair}(x).$$

Il faudrait donc raffiner l'analyse des déterminants de quelque façon, mais nous ne le ferons pas ici.[57]. Dans la langue mathématique, nous avons la quantification avec des variables explicites, dont l'interprétation ne pose pas de problèmes.

Nous avons fait une distinction entre les **noms propres collectifs** et les **syntagmes nominaux collectifs**. Ceux-ci sont utilisés comme sujets des verbes collectifs ; ceux-là, une catégorie plus restreinte, sont utilisés comme des arguments des fonctions individuelles. Tous les deux peuvent être formés comme des conjonctions de noms propres. Le

[56] En anglais, cette opération s'appelle *raising*. Elle est connue, surtout, des œuvres de Montague. Mais elle est déjà implicite dans la discussion de Frege (1879, § 10), qui constate que l'expression $\Phi(A)$ peut aussi être conçue comme une fonction de l'argument Φ.

[57] Si on analyse les verbes négatifs comme arguments des syntagmes nominaux, la portée du quantificateur doit être plus large que la portée de la négation. Le même phénomène est illustré par les noms coordonnés : l'interprétation de 3 *et* 5 *ne sont pas pairs* est $\sim$Pair(3) & $\sim$Pair(5), sans ambiguïté.

nom propre collectif est un complexe de deux noms, qui peuvent être de types différents. Le syntagme nominal collectif est construit d'une liste d'au moins trois noms, qui doivent être du même type.[58]

PNC $\rightarrow$ PN *et* PN

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad a : \text{PN}(\Gamma, A) \quad b : \text{PN}(\Gamma, B)}{\begin{array}{l}\text{CouplePN}(\Gamma, A, B, a, b) : \text{PNC}(\Gamma, A, B) \\ \text{CouplePN}(\Gamma, A, B, a, b)^\star = (a^\star, b^\star) \\ \text{CouplePN}(\Gamma, A, B, a, b)^\circ(r) = \text{etcoll}(r, [(\lambda y)a^\circ(y), (\lambda y)b^\circ(y)]) \\ \text{gen}(\text{CouplePN}(\Gamma, A, B, a, b)) = \text{collgenre}([\text{gen}(a), \text{gen}(b)])\end{array}}$$

SNC $\rightarrow$ PN, ... PN *et* PN

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad l : \text{list}_3(\text{PN}(\Gamma, A))}{\begin{array}{l}\text{CollPN}(\Gamma, A, l) : \text{SNC}(\Gamma, A) \\ \text{CollPN}(\Gamma, A, l)^\star = (Y)(\forall x, y \in l)Y(x^\star, y^\star) \\ \text{CollPN}(\Gamma, A, a, b)^\circ(r) = \text{etcoll}(r, \text{map}_3((x)(\lambda y)x^\circ(y), l)) \\ \text{gen}(\text{CollPN}(\Gamma, A, a, b)) = \text{collgenre}(\text{map}_3((x)\text{gen}(x), l))\end{array}}$$

SNC $\rightarrow$ PNC

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad c : \text{PNC}(\Gamma, A, B)}{\begin{array}{l}\text{MontPNC}(\Gamma, A, B, c) : \text{SNC}(\Gamma, A, B) \\ \text{MontPNC}(\Gamma, A, B, c)^\star = (Y)Y(p(c^\star), q(c^\star)) \\ \text{MontPNC}(\Gamma, A, B, c)^\circ(r) = c^\circ(r) \\ \text{gen}(\text{MontPNC}(\Gamma, A, B, c)) = \text{gen}(c)\end{array}}$$

Par exemple, l'énoncé

$$1, 2 \textit{ et } 3 \textit{ sont distincts}$$

résulte de l'arbre

$$\text{PrédVC}(\text{CollPN}([1, 2, 3]), \text{PrédAC}(\text{être}(\text{vrai}), \text{distincts}))$$

qui est interprété

$$D(1, 2) \;\&\; D(1, 3) \;\&\; D(2, 3).$$

Cette **conjonction collective** de termes doit être bien distinguée de la **conjonction distributive**, que nous allons discuter dans la section 2.12, comme un cas spécial de la coordination. Notons que la disjonction *ou* n'a que l'usage distributif.

Les **fonctions individuelles**, telles que

carré : $\text{F1}(\Gamma, R, R, de)$,

valeur : $\text{F2}(\Gamma, R \rightarrow R, R, R, de, à)$,

somme : $\text{FC}(\Gamma, R, R, R, de)$,

[58]Les verbes collectifs d'un type $\text{VC}(\Gamma, A, B)$ où A et B sont des types distincts sont assez rares. Il y a, par exemple, le prédicat hilbertien *le point a et la droite l sont incidents*. On se demande comment l'application de ce prédicat à plus que deux noms serait interprétée ; une interprétation serait que chaque point mentionné est incident avec chaque droite mentionnée. Nous avons laissé de telles constructions de côté.

sont appliquées à des nom propres pour produire des noms propres, selon les règles suivantes.

PN $\rightarrow$ *le* F1(r) r PN

$$\frac{\Gamma : \text{cont} \quad A, D : \text{set}/\Gamma \quad r : \text{préfixe} \quad f : \text{F1}(\Gamma, A, D, r) \quad a : \text{PN}(\Gamma, A)}{\begin{array}{l}\text{ApplF1}(\Gamma, A, D, r, f, a) : \text{PN}(\Gamma, D) \\ \text{ApplF1}(\Gamma, A, D, r, f, a)^\star = f^\star(a^\star) \\ \text{ApplF1}(\Gamma, A, D, r, f, a)^\circ(t) = le(t, \text{gen}(f), \text{sg}, f^\circ(\text{sg}))\ a^\circ(r) \\ \text{gen}(\text{ApplF1}(\Gamma, A, D, r, f, a)) = \text{gen}(f)\end{array}}$$

F1(t) $\rightarrow$ F2(r, t) r PN

$$\frac{\Gamma : \text{cont} \quad A, B, D : \text{set}/\Gamma \quad r, t : \text{préfixe} \quad f : \text{F2}(\Gamma, A, B, D, r, t) \quad a : \text{PN}(\Gamma, A)}{\begin{array}{l}\text{ApplF2}(\Gamma, A, B, D, r, t, f, a) : \text{F1}(\Gamma, B, D, t) \\ \text{ApplF2}(\Gamma, A, B, D, r, t, f, a)^\star = (y) f^\star(a^\star, y) \\ \text{ApplF2}(\Gamma, A, B, D, r, t, f, a)^\circ(n) = f^\circ(n)\ a^\circ(r) \\ \text{gen}(\text{ApplF2}(\Gamma, A, B, D, r, t, f, a)) = \text{gen}(f)\end{array}}$$

PN $\rightarrow$ FC(r) r PNC

$$\frac{\Gamma : \text{cont} \quad A, B, D : \text{set}/\Gamma \quad r : \text{préfixe} \quad f : \text{FC}(\Gamma, A, B, D, r) \quad c : \text{PNC}(\Gamma, A, B)}{\begin{array}{l}\text{ApplFC}(\Gamma, A, B, D, r, f, c) : \text{PN}(\Gamma, D) \\ \text{ApplFC}(\Gamma, A, B, D, r, f, c)^\star = (y) f^\star(p(c^\star), q(c^\star)) \\ \text{ApplFC}(\Gamma, A, B, D, r, f, c)^\circ(t) = le(t, \text{gen}(f), \text{sg}, f^\circ(\text{sg}))\ c^\circ(r) \\ \text{gen}(\text{ApplFC}(\Gamma, A, B, D, r, f, c)) = \text{gen}(f)\end{array}}$$

Un exemple qui montre toutes ces règles: l'arbre

ApplFC(*de*, somme,
CouplePN(ApplF1(*de*, carré, $V(x)$), ApplF2(*de*, *à*, valeur, $V(f), V(y)$))),

qui est interprété $x^2 + f(y)$, se linéarise en

la somme du carré de x et de la valeur de f à y.

(La formation des noms propres à partir des symboles de variables, désignée ici par l'opérateur V, sera expliquée dans la section 3.4.)

Dans les règles ApplF1 et ApplF2, il y a une possibilité d'affaiblir le type des arguments du nom propre au syntagme nominal, pour former des expressions telles que

les carrés de 3 *et de* 4,

interprétée

$$(Y)(Y(3^2)\ \&\ Y(4^2)).$$

Mais en même temps, le type de la conclusion devient SN aussi, de sorte qu'on ne peut pas dire que la règle devienne plus forte.

Une fonction collective pourrait être appliquée à une liste de noms propres pour produire un nom propre, par exemple,

la somme de 2, *de* 3 *et de* 4,

pourvu que l'opération désignée soit associative. Autrement, le résultat serait incompréhensible:

la différence de 5, *de* 2 *et de* 1.

2.8 *Les présuppositions des fonctions individuelles*

Les expressions

la fonction inverse de f,

la dérivée de f au point x,

la point d'intersection des droites l et m,

sont, superficiellement, formées par les règles d'application présentées dans la section précédente. Mais ces règles n'expriment pas les **présuppositions** propres à chacune de ces formes: dans le premier exemple, il faut que f soit bijective; dans la deuxième, f doit être dérivable en x; dans la troisième, il faut que l et m s'intersectent. De telles **fonctions conditionnées** sont si fréquentes qu'il vaut mieux introduire des catégories générales, correspondant à toutes les catégories de fonctions:

catégorie	condition	interprétée
$\mathrm{F1C}(\Gamma, A, C, D, r)$	$C : (A)\mathrm{prop}/\Gamma$	$(x : A)(C(x))D/\Gamma$
$\mathrm{F2C}(\Gamma, A, B, C, D, r, t)$	$C : (A)(B)\mathrm{prop}/\Gamma$	$(x : A)(y : B)(C(x,y))D/\Gamma$
$\mathrm{FCC}(\Gamma, A, B, C, D, r)$	$C : (A)(B)\mathrm{prop}/\Gamma$	$(x : A)(y : B)(C(x,y))D/\Gamma$

Ainsi nous pouvons catégoriser

inverse : $\mathrm{F1C}(\Gamma, R \to R, \mathrm{Bij}, R, de)$,

dérivée : $\mathrm{F2C}(\Gamma, R \to R, R, \mathrm{Dér}, R, de, \grave{a})$,

point-d'intersection : $\mathrm{FCC}(\Gamma, \mathrm{Ln}, \mathrm{Ln}, \mathrm{Int}, \mathrm{Pt}, de)$,

où les définitions des prédicats

$$\mathrm{Bij} : (R \to R)\mathrm{prop},\ \mathrm{Dér} : (R \to R)(R)\mathrm{prop},\ \mathrm{Int} : (\mathrm{Ln})(\mathrm{Ln})\mathrm{prop},$$

seront familières.

Les catégories de fonctions sans présuppositions sont définissables en termes de ces catégories-ci, en instanciant la condition C par la fonction propositionnelle qui est uniformément vraie.

Les règles d'application sont modifiées par l'introduction d'un argument sémantique qui prouve la présupposition:[59]

$$\frac{\begin{array}{l}\Gamma : \mathrm{cont}\quad A : \mathrm{set}/\Gamma\quad C : (A)\mathrm{prop}/\Gamma\quad D : \mathrm{set}/\Gamma\quad r : \text{préfixe}\\ f : \mathrm{F1C}(\Gamma, A, C, D, r)\quad a : \mathrm{PN}(\Gamma, A)\quad c : C(a^\star)\end{array}}{\begin{array}{l}\mathrm{ApplF1C}(\Gamma, A, C, D, r, f, a, c) : \mathrm{PN}(\Gamma, D)\\ \mathrm{ApplF1C}(\Gamma, A, C, D, r, f, a, c)^\star = f^\star(a^\star, c)\\ \mathrm{ApplF1C}(\Gamma, A, C, D, r, f, a, c)^\circ(t) = le(t, \mathrm{gen}(f), \mathrm{sg}, f^\circ(\mathrm{sg}))\ a^\circ(r)\\ \mathrm{gen}(\mathrm{ApplF1C}(\Gamma, A, C, D, r, f, a, c)) = \mathrm{gen}(f)\end{array}}$$

$$\frac{\begin{array}{l}\Gamma : \mathrm{cont}\quad A, B : \mathrm{set}/\Gamma\quad C : (A)(B)\mathrm{prop}/\Gamma\quad D : \mathrm{set}/\Gamma\quad r, t : \text{préfixe}\\ f : \mathrm{F2C}(\Gamma, A, B, C, D, r, t)\quad a : \mathrm{PN}(\Gamma, A)\end{array}}{\begin{array}{l}\mathrm{ApplF2C}(\Gamma, A, B, C, D, r, t, f, a) : \mathrm{F1C}(\Gamma, B, (y)C(a^\star, y), D, t)\\ \mathrm{ApplF2C}(\Gamma, A, B, C, D, r, t, f, a)^\star = (y)(z)f^\star(a^\star, y, z)\\ \mathrm{ApplF2C}(\Gamma, A, B, C, D, r, t, f, a)^\circ(n) = f^\circ(n)\ a^\circ(r)\\ \mathrm{gen}(\mathrm{ApplF2C}(\Gamma, A, B, C, D, r, t, f, a)) = \mathrm{gen}(f)\end{array}}$$

[59] Notons que cette modification n'est pas visible dans les règles syntagmatiques. L'opérateur ApplF2C n'a pas cet argument sémantique, mais il communique la condition à la fonction du type F1C formée.

Γ : cont A, B : set/Γ C : $(A)(B)$prop/Γ D : set/Γ r : préfixe
f : FCC(Γ, A, B, D, r) c : PNC(Γ, A, B) d : $C(p(c^\star), q(c^\star))$

ApplFCC($\Gamma, A, B, C, D, r, f, c, d$) : PN($\Gamma, D$)
ApplFCC($\Gamma, A, B, C, D, r, f, c, d)^\star$ = $(y)f^\star(p(c^\star), q(c^\star), d)$
ApplFCC($\Gamma, A, B, C, D, r, f, c, d)^\circ(t)$ = $le(t, \text{gen}(f), \text{sg}, f^\circ(\text{sg}))\ c^\circ(r)$
gen(ApplFCC($\Gamma, A, B, C, D, r, f, c, d$)) = gen($f$)

2.9 *La proposition relative*

Une **proposition relative** est une phrase introduite par un **pronom relatif**, qui sert à modifier un nom commun ou un syntagme nominal. Nous introduisons la catégorie

$$\text{PR}(\Gamma, A)$$

des propositions relatives, interprétée de la même façon que les adjectifs à une place,

$$(A)\text{prop}/\Gamma.$$

La linéarisation dépend du genre, du nombre, et du mode.

Une proposition relative peut être formée d'une phrase à laquelle manque un argument — c'est le pronom relatif qui, pour ainsi dire, prend la place de l'argument. Les catégories correspondantes sont V1 et $\text{V}^1/_2$.[60] Par exemple, la phrase sans complément,

l est perpendiculaire,

de la catégorie $\text{V}^1/_2(\Gamma, \text{Ln}, \grave{a})$, et avec l'interprétation

$$(y)\text{Perp}(l, y) : (\text{Ln})\text{prop},$$

devient la proposition relative

à laquelle l est perpendiculaire

de la catégorie PR(Γ, Ln, $\grave{a}$), avec la même interprétation. Mais le pronom relatif peut aussi changer le type du prédicat : la relative

au grand-axe de laquelle l est perpendiculaire

est définie pour les ellipses. Elle est donc de la catégorie PR(Γ, Ell, $\grave{a}$), et interprétée

$$(y)\text{Perp}(l, \text{ga}(y)) : (\text{Ell})\text{prop}.$$

Ainsi un pronom relatif doit être interprétée comme une opération qui prend une fonction propositionnelle comme argument et retourne une fonction propositionnelle dont l'argument peut être d'un autre type. La catégorie des pronoms relatifs dépend ainsi de deux domaines, mais aussi du contexte, comme le prouve le pronom

[60] La méthode des grammairiens transformationnels pionniers pour obtenir une telle phrase sans argument était d'enlever n'importe quel élément nominal d'une phrase complète, mais cette méthode a été trouvée trop générale. Nous suivons, pour ainsi dire, une méthode contraire : partir d'un système délimité de catégories qui correspondent aux phrases sans arguments. Cette méthode a été développée dans la grammaire syntagmatique généralisée ; cf. Gazdar *et al.* (1985, chapitre 7), et Abeillé (1993, section 2.5).

la valeur de laquelle à x,

qui contient une variable.

En français, il y a un système très net de pronoms relatifs basé sur le pronom *lequel.* Nous introduisons la catégorie

$$\mathrm{Lq}(\Gamma, A, B)$$

pour ce système de pronoms relatifs, interprétée

$$((A)\mathrm{prop})(B)\mathrm{prop}/\Gamma.$$

La linéarisation dépend des paramètres de genre, de nombre et de préfixe. Puisque le pronom relatif peut fonctionner comme le sujet d'un verbe, il lui faut communiquer un genre et un nombre à celui-ci. Comme nous verrons, ces paramètres dépendent et du pronom lui-même et de l'antécédent de la relative (c.-à-d. du nom modifié par elle).

Les pronoms *qui*, *que* et *dont* sont utilisés d'une façon beaucoup plus particulière, en dehors de ce système. Nous les traiterons directement comme des constructeurs de propositions relatives, sans essayer de définir une catégorie de pronoms à laquelle ils appartiennent.[61]

Deux règles suffisent à définir la catégorie Lq :

Lq $\rightarrow$ *lequel*

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{set}/\Gamma}{\begin{array}{l}\mathrm{lequel}(\Gamma, A) : \mathrm{Lq}(\Gamma, A, A) \\ \mathrm{lequel}(\Gamma, A)^{\star} = (Y)Y \\ \mathrm{lequel}(\Gamma, A)^{\circ}(g, n, r) = \mathit{lequel}(g, n, r) \\ \mathrm{gen}(\mathrm{lequel}(\Gamma, A), g) = g\end{array}}$$

Lq $\rightarrow$ *le* Fl(r) r Lq

$$\frac{\Gamma : \mathrm{cont} \quad A, B, C : \mathrm{set}/\Gamma \quad r : \text{préfixe} \quad f : \mathrm{Fl}(\Gamma, A, B, r) \quad L : \mathrm{Lq}(\Gamma, A, C)}{\begin{array}{l}\mathrm{LqFl}(\Gamma, A, B, C, r, f, L) : \mathrm{Lq}(\Gamma, B, C) \\ \mathrm{LqFl}(\Gamma, A, B, C, r, f, L)^{\star} = (Y)(z)L^{\star}((x)Y(f^{\star}(x)), z) \\ \mathrm{LqFl}(\Gamma, A, B, C, r, f, L)^{\circ}(g, n, t) = \mathit{le}(t, \mathrm{gen}(f), n, f^{\circ}(n))\ L^{\circ}(g, n, r) \\ \mathrm{gen}(\mathrm{LqFl}(\Gamma, A, B, C, r, f, L), g) = \mathrm{gen}(f)\end{array}}$$

C'est la seconde règle qui est capable de changer le domaine du prédicat.[62] [63] En même temps, elle peut changer le genre communiqué à lui. Par exemple, on dit

[61] Il y a une exception à la grande régularité des pronoms basés sur *lequel*: «comme objet direct, *lequel* est un archaïsme assez rare» (Grevisse, § 692, *c*). Nous ignorerons cette exception.

[62] Il n'est pas possible que f ait une présupposition, à moins qu'il soit garanti que tous les éléments du domaine antécédent la satisfont. Ainsi l'énoncé

fonction l'inverse de laquelle est croissante

n'est pas bien formé. L'expression

bijection l'inverse de laquelle est croissante

pourrait l'être, mais pour l'analyser, il nous faudrait introduire une catégorie de propositions relatives dépendantes d'une présupposition.

[63] C'est ici que nous avons, pour la première fois, besoin du paramètre de nombre des fonctions : nous voulons former des expressions telles que *nombres les carrés desquels sont impairs.*

ellipse qui est verticale,

ellipse le grand-axe de laquelle est vertical,

bien que l'antécédent soit le même dans les deux expressions, *ellipse.*[64] [65]

Avec nos prédicats à deux places au maximum, nous avons deux règles pour la formation des propositions relatives :

PR $\rightarrow$ Lq V1

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad L : \text{Lq}(\Gamma, A, B) \quad F : \text{V1}(\Gamma, A)}{\begin{array}{l}\text{RelV1}(\Gamma, A, B, L, F) : \text{PR}(\Gamma, B) \\ \text{RelV1}(\Gamma, A, B, L, F)^\star = (y)L^\star(F^\star, y) \\ \text{RelV1}(\Gamma, A, B, L, F)^\circ(g, n, m) = L^\circ(g, n, -)\ F^\circ(\text{gen}(L, g), n, m)\end{array}}$$

PR $\rightarrow$ Lq $\text{V}^1/2$

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad r : \text{préfixe} \quad L : \text{Lq}(\Gamma, A, B) \quad F : \text{V}^1/2(\Gamma, A, r)}{\begin{array}{l}\text{RelV}^1/2(\Gamma, A, B, r, L, F) : \text{PR}(\Gamma, B) \\ \text{RelV}^1/2(\Gamma, A, B, r, L, F)^\star = (y)L^\star(F^\star, y) \\ \text{RelV}^1/2(\Gamma, A, B, r, L, F)^\circ(g, n, m) = L^\circ(g, n, r)\ F^\circ(m)\end{array}}$$

Les pronoms *qui* et *que* sont analogues aux règles RelV1 et $\text{RelV}^1/2$, mais ils ne sont pas capables de changer le domaine, et *que* ne s'applique pas aux verbes à compléments prépositionnels.

PR $\rightarrow$ *qui* V1

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad F : \text{V1}(\Gamma, A)}{\begin{array}{l}\text{qui}(\Gamma, A, F) : \text{PR}(\Gamma, A) \\ \text{qui}(\Gamma, A, F)^\star = F^\star \\ \text{qui}(\Gamma, A, F)^\circ(g, n, m) = \textit{qui}\ F^\circ(g, n, m)\end{array}}$$

PR $\rightarrow$ *que* $\text{V}^1/2(-)$

$$\frac{\Gamma : \text{cont} \quad B : \text{set}/\Gamma \quad F : \text{V}^1/2(\Gamma, B, -)}{\begin{array}{l}\text{que}(\Gamma, B) : \text{PR}(\Gamma, B) \\ \text{que}(\Gamma, B)^\star = F^\star \\ \text{que}(\Gamma, B)^\circ(g, n, m) = \textit{que}(F^\circ(m))\end{array}}$$

Le pronom *dont* fonctionne à peu près comme *duquel*: il peut former un pronom complexe avec une fonction de la catégorie F1, et ainsi changer le domaine ; il peut servir comme complément prépositionnel si la préposition est *de.*

[64] Grevisse dit : « *Lequel* varie en genre et en nombre en fonction de son antécédent et communique ce genre et ce nombre aux mots qui s'accordent avec lui » (§ 680). Cette explication n'est pas tout à fait exacte quant au genre : dans notre seconde regle, il y a deux genres en jeu, le genre gen(f) inhérent de la fonction f, et le genre g, paramètre de la construction. Le genre de l'antécédent est g mais le genre communiqué est gen(f).

[65] Il en est de même à propos du nombre : le nombre grammatical communiqué par le pronom relatif peut différer du nombre de l'antécédent, comme dans l'exemple *nombre les facteurs duquel sont impairs.* Mais nous n'avons pas formulé de règles qui introduisent des fonctions avec cet effet.

PR $\rightarrow$ *dont le* F1(de) V1

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad f : \text{F1}(\Gamma, A, B, de) \quad F : \text{V1}(\Gamma, B)}{\begin{array}{l}\text{dontF1V1}(\Gamma, A, B, f, F) : \text{PR}(\Gamma, A) \\ \text{dontF1V1}(\Gamma, A, B, f, F)^\star = (x)F^\star(f^\star(x)) \\ \text{dontF1V1}(\Gamma, A, B, f, F)^\circ(g, n, m) = \textit{dont le}(-, \text{gen}(f), n, f^\circ(n))\ F^\circ(\text{gen}(f), n, m)\end{array}}$$

PR $\rightarrow$ *dont le* F1(de) V$^1/_2(-)$

$$\frac{\Gamma : \text{cont} \quad A, B : \text{set}/\Gamma \quad f : \text{F1}(\Gamma, A, B, de) \quad F : \text{V}^1/_2(\Gamma, B, -)}{\begin{array}{l}\text{dontF1V}^1/_2(\Gamma, A, B, f, F) : \text{PR}(\Gamma, A) \\ \text{dontF1V}^1/_2(\Gamma, A, B, f, F)^\star = (x)F^\star(f^\star(x)) \\ \text{dontF1V}^1/_2(\Gamma, A, B, f, F)^\circ(g, n, m) = \textit{dont le}(-, \text{gen}(f), n, f^\circ(n))\ F^\circ(m)\end{array}}$$

PR $\rightarrow$ *dont* V$^1/_2$(de)

$$\frac{\Gamma : \text{cont} \quad B : \text{set}/\Gamma \quad F : \text{V}^1/_2(\Gamma, B, de)}{\begin{array}{l}\text{dontV}^1/_2(\Gamma, B) : \text{PR}(\Gamma, B) \\ \text{dontV}^1/_2(\Gamma, B)^\star = F^\star \\ \text{dontV}^1/_2(\Gamma, B)^\circ(g, n, m) = \textit{dont}\ F^\circ(m)\end{array}}$$

Enfin, il y a un pronom très fréquent dans les textes mathématiques, **tel que**. Sa syntaxe est simple : il s'attache à une phrase complète. Mais le contexte où cette phrase est formée est étendu par une variable du type correspondant à l'antécédent de la relative.

PR $\rightarrow$ *tel que* P

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad B : \text{P}((\Gamma,\ x : A))}{\begin{array}{l}\text{tel-que}(\Gamma, A, B) : \text{PR}(\Gamma) \\ \text{tel-que}(\Gamma, A, B)^\star = (x)B^\star \\ \text{tel-que}(\Gamma, A, B)^\circ(g, n, m) = \textit{tel}(g, n)\ \textit{que}(B^\circ(m))\end{array}}$$

2.10 *La modification du nom commun*

Étant donné que les adjectifs à une place et les propositions relatives sont interprétés comme des prédicats à une place, et que les noms communs sont interprétés comme des domaines, la **modification** du nom commun peut être définie comme la formation d'un ensemble Σ. Par exemple, les modifiés

nombre pair,

nombre qui est pair,

sont tous les deux interprétés comme

$$(\Sigma x : N)\text{Pair}(x).$$

Ainsi un nombre pair est un nombre avec une preuve qu'il est pair. La preuve est appelée le **témoin** dans la mathématique constructive.[66]

[66] Cette interprétation est présentée sous le titre « the notion of such that » par Martin-Löf (1984, pp. 53–54).

Donc les règles de modification sont

CN $\rightarrow$ CN PR

$$\frac{\Gamma : \text{cont} \quad A : \text{CN}(\Gamma) \quad B : \text{PR}(\Gamma, A^\star)}{\begin{array}{l}\text{ModPR}(\Gamma, A, B) : \text{CN}(\Gamma) \\ \text{ModPR}(\Gamma, A, B)^\star = (\Sigma x : A^\star)B^\star \\ \text{ModPR}(\Gamma, A, B)^\circ(n) = A^\circ(n)\ B^\circ(\text{gen}(A), n, \text{ind}) \\ \text{gen}(\text{ModPR}(\Gamma, A, B)) = \text{gen}(A)\end{array}}$$

CN $\rightarrow$ CN A1

$$\frac{\Gamma : \text{cont} \quad A : \text{CN}(\Gamma) \quad B : \text{A1}(\Gamma, A^\star)}{\begin{array}{l}\text{ModA1}(\Gamma, A, B) : \text{CN}(\Gamma) \\ \text{ModA1}(\Gamma, A, B)^\star = (\Sigma x : A^\star)B^\star \\ \text{ModA1}(\Gamma, A, B)^\circ(n) = A^\circ(n)\ B^\circ(\text{gen}(A), n) \\ \text{gen}(\text{ModA1}(\Gamma, A, B)) = \text{gen}(A)\end{array}}$$

Cette construction est utilisée, par exemple, pour la quantification sur les éléments qui ont la propriété modifiante. La proposition relative est toujours mise à l'indicatif. Plus tard, dans la section 3.6, nous présenterons quelques constructions où la relative est au subjonctif.

La règle de modification relative que nous venons de présenter est celle de la modification **déterminative**, ou **restrictive**, où la relative «restreint l'extension du terme qu'elle accompagne (la suppression de la relative modifierait profondément le message)» (Grevisse, § 1059). Il y a aussi de la modification **non déterminative**, ou **explicative**, ou bien **appositive**, où la relative «ne restreint pas l'extension du terme... on peut la rapprocher de l'épithète détaché» (*ibid.*).

La distinction entre la modification déterminative et appositive est assez claire, parce qu'il y a une différence syntaxique: ce qui est modifié est un nom commun dans le premier cas, et un syntagme nominal dans le second cas. Dans le second cas, la propriété modifiante est présupposée, ce qui explique pourquoi «la suppression de la relative ne modifierait pas vraiment le message» (Grevisse, § 1059). Ainsi l'interprétation du syntagme nominal modifié est la même que l'interprétation du syntagme originel: la propriété modifiante n'entre dans la construction que comme un des arguments présupposés. La relative est séparée par une paire de virgules.

SN $\rightarrow$ SN, PR,

$$\frac{\Gamma : \text{cont} \quad A : \text{set}/\Gamma \quad Q : \text{SN}(\Gamma, A) \quad B : \text{PR}(\Gamma, A) \quad c : Q^\star(B^\star)}{\begin{array}{l}\text{ApposPR}(\Gamma, A, Q, B, c) : \text{SN}(\Gamma, A) \\ \text{ApposPR}(\Gamma, A, Q, B, c)^\star = Q^\star \\ \text{ApposPR}(\Gamma, A, Q, B, c)^\circ(r) = Q^\circ(r),\ B^\circ(\text{gen}(Q), \text{num}(Q), \text{ind}), \\ \text{gen}(\text{ApposPR}(\Gamma, A, Q, B, c)) = \text{gen}(Q) \\ \text{num}(\text{ApposPR}(\Gamma, A, Q, B, c)) = \text{num}(Q)\end{array}}$$

L'usage typique de ApposPR est de faire un rappel d'un fait qui est déjà connu. Par exemple, dans

l'inverse de la fonction f, qui est bijective, est...

la relative appositive justifie, pour le lecteur, la formation de l'inverse de f.

2.11 *La composition des fonctions*

Dans la théorie des types, les fonctions peuvent être composées, si les types se conforment, selon la règle

$$\frac{g : (\beta)\gamma \quad f : (\alpha)\beta}{g \circ f = (x)g(f(x)) : (\alpha)\gamma}.$$

Dans la plupart des constructions grammaticales, les fonctions composées ne sont pas reconnues comme constituants : par exemple, l'expression

le successeur du carré de 3

peut être formée, simplement, par deux applications successives de ApplF1.

Mais pour former le nom modifié

nombre dont le successeur du carré est premier,

il nous faut pouvoir former la **fonction composée** du successeur et du carré, ce qui nous est fournie par la règle

$\mathrm{F1}(r) \rightarrow \mathrm{F1}(t)\ t\ \mathrm{F1}(r)$

$$\frac{\Gamma : \mathrm{cont} \quad A, B, C : \mathrm{set}/\Gamma \quad r, t : \text{préfixe} \quad g : \mathrm{F1}(\Gamma, B, C, t) \quad f : \mathrm{F1}(\Gamma, A, B, r)}{\begin{array}{l}\mathrm{CompF1}(\Gamma, A, B, C, r, t, g, f) : \mathrm{F1}(\Gamma, A, C, r) \\ \mathrm{CompF1}(\Gamma, A, B, C, r, t, g, f)^\star = (x)g^\star(f^\star(x)) \\ \mathrm{CompF1}(\Gamma, A, B, C, r, t, g, f)^\circ(n) = g^\circ(n)\ le(t, \mathrm{gen}(f), n, f^\circ(n)) \\ \mathrm{gen}(\mathrm{CompF1}(\Gamma, A, B, C, r, t, g, f)) = \mathrm{gen}(g)\end{array}}$$

Dans la linguistique générale, la composition des fonctions est employée en analyse des **dépendances non bornées** dans les constructions relatives, c.-à-d. des propositions relatives où l'argument manquant de la proposition peut apparaître à n'importe quelle profondeur dans l'arbre. Par exemple, l'expression

ellipse dont l coupe le grand-axe

est formée par la composition d'un V2 avec un F1,[67] [68]

$\mathrm{V2}(r) \rightarrow \mathrm{V2}(t)\ t\ \mathrm{F1}(r)$

$$\frac{\Gamma : \mathrm{cont} \quad A, B, C : \mathrm{set}/\Gamma \quad r, t : \text{préfixe} \quad F : \mathrm{V2}(\Gamma, A, B, t) \quad f : \mathrm{F1}(\Gamma, C, B, r)}{\begin{array}{l}\mathrm{CompV2}(\Gamma, A, B, C, r, t, F, f) : \mathrm{V2}(\Gamma, A, C, r) \\ \mathrm{CompV2}(\Gamma, A, B, C, r, t, F, f)^\star = (x)(z)F^\star(x, f^\star(z)) \\ \mathrm{CompV2}(\Gamma, A, B, C, r, t, F, f)^\circ(g, n, m) = F^\circ(g, n, m)\ le(t, \mathrm{gen}(f), n, f^\circ(n))\end{array}}$$

[67] C'est la première de nos règles qui permette la formation des V2 dépendants d'un contexte : la fonction peut contenir une variable, comme dans l'exemple *point par lequel m coupe la parallèle à l*.

[68] Le traitement des dépendances non bornées par la composition des fonctions est standard dans les grammaires syntagmatiques généralisées et dans les grammaires catégorielles ; cf. Abeillé (1993) et, surtout, Steedman (1988).

2.12 *La coordination*

La catégorie des **conjonctions**, C1, est interprétée comme le type des connecteurs à deux places,

$$\mathrm{C1}^\star \;=\; (\mathrm{prop})(\mathrm{prop})\mathrm{prop}.$$

Les conjonctions sont linéarisées sans paramètres. Les plus fréquentes seront *et* et *ou*, interprétées respectivement comme & et $\vee$.

L'application naturelle des conjonctions va aux phrases, mais il est possible de **coordonner** des expressions de presque n'importe quelle catégorie, selon le schéma

$$\alpha \;\to\; \alpha,\; \ldots\; \mathrm{C1}\;\alpha.$$

Mais nous ne pouvons pas adopter ce schéma dans toute sa généralité, parce qu'il nous faut interpréter toutes les expressions formées. Dans ce cas-ci, il y a une interprétation naturelle pour toutes les catégories interprétées comme des types de la forme

$$(x_1 \,:\, \alpha_1)\cdots(x_n \,:\, \alpha_n)\mathrm{prop},$$

à savoir, la conjonction des termes coordonnés, avec les abstractions nécessaires. Par exemple, la conjonction de m verbes intransitifs $F_1, \ldots, F_m$ avec *ou* est interprétée

$$(x)(F_1^\star(x) \vee \ldots \vee F_m^\star(x)).$$

Les paramètres de linéarisation sont distribués à chaque terme coordonné.

Pour formaliser la règle de coordination, nous utilisons les listes d'au moins deux éléments. L'interprétation de la conjonction est définie par l'application de l'interprétation à chacune des expressions de la liste. La linéarisation est analogue.

$$\frac{C \,:\, \mathrm{C1} \quad L \,:\, \mathrm{list}_2(\alpha)}{\begin{array}{l}\mathrm{Conj}\alpha(L) \,:\, \alpha \\ \mathrm{Conj}\alpha(L)^\star \;=\; (x_1)\cdots(x_n)\mathrm{listconn}(C^\star, \mathrm{map}_2((X)X^\star(x_1,\ldots,x_n), L)) \\ \mathrm{Conj}\alpha(L)^\circ(p_1,\ldots,p_k) \;=\; \mathrm{conjform}(C^\circ, \mathrm{map}_2((X)X^\circ(p_1,\ldots,p_k), L))\end{array}}$$

Ainsi la coordination peut être définie pour les phrases et pour toutes les catégories interprétés comme des types de prédicats — pour V1, A1, $\mathrm{V}^1/_2$, PR, V2, A2, Aq, VC et AC. Par exemple, les verbes à deux places sont coordonnés selon la règle

$$\frac{\Gamma \,:\, \mathrm{cont} \quad A, B \,:\, \mathrm{set}/\Gamma \quad r \,:\, \mathrm{préfixe} \quad L \,:\, \mathrm{list}_2(\mathrm{V2}(\Gamma, A, B, r))}{\begin{array}{l}\mathrm{ConjV2}(\Gamma, A, B, r, L) \,:\, \mathrm{V2}(\Gamma, A, B, r) \\ \mathrm{ConjV2}(\Gamma, A, B, r, L)^\star \;=\; (x)(y)\mathrm{listconn}(C^\star, \mathrm{map}_2((X)X^\star(x,y), L)) \\ \mathrm{ConjV2}(\Gamma, A, B, r, L)^\circ(g,n,m) \;=\; \mathrm{conjform}(C^\circ, \mathrm{map}_2((X)X^\circ(g,n,m))\end{array}}$$

Sémantiquement, il est en principe aussi possible de coordonner des déterminants, des pronoms relatifs et des syntagmes nominaux:[69]

[69] La coordination des syntagmes nominaux pose le problème de la définition des traits inhérents. Le genre a une définition familière par le genre collectif: si tous les syntagmes coordonnés sont féminins, leur conjonction est féminine; autrement, elle est masculine. Mais le nombre dépend de la conjonction: si elle est *et*, le nombre est pluriel; si elle est *ou*, le nombre est singulier si tous les syntagmes coordonnés sont singuliers, autrement pluriel. Pour exprimer cette variation sans violer la compositionnalité, nous pouvons définir le nombre comme un trait inhérent des conjonctions, ou simplement introduire deux règles séparées. Un autre problème est la distribution du préfixe. Selon nos règles, toutes les prépositions sont distribuées à tous les éléments de la coordination. Il y a quelques distinctions qu'on pourrait faire; cf. Grevisse (§ 995), selon lequel toutes les prépositions peuvent se répéter, sauf *entre*. Mais cette exception ne concerne que les constructions qui sont logiquement des syntagmes collectifs, plutôt que coordonnés; cf. sections 2.1 et 2.7.

tous ou presque tous *les points de A,*

la personne **sur laquelle et avec laquelle** *vous parlez,*

tous les nombres ou presque tous les nombres.

Par voie de conséquence, nous pouvons coordonner des noms propres en les transformant en des syntagmes nominaux par l'opération de montée. Mais l'expression

$$2,\ 3\ \textit{ou}\ 4,$$

n'est pas une instance du schéma intial avec α = PN, parce que le résultat n'est pas lui-même un nom propre.

Les **conjonctions répétées**, telles que *ou...ou*, *et...et*, *soit...soit* sont utilisées d'une façon semblable.[70] Elles sont intéressantes pour la langue mathématique, parce qu'elles ne produisent pas tant d'ambiguïtés. Par exemple, l'énoncé

$$2\ \textit{et}\ 3\ \textit{ou}\ 4,$$

est ambigu, mais les deux alternatives peuvent être exprimées par les énoncés univoques

$$\textit{ou}\ 2\ \textit{et}\ 3\ \textit{ou}\ 4,\ 2\ \textit{et ou}\ 3\ \textit{ou}\ 4.$$

Soit CR la catégorie des conjonctions répétées, interprétée (prop)(prop)prop. Le schéma de coordination pour CR est

$\alpha \to \mathrm{CR}\,\alpha \ldots \mathrm{CR}\,\alpha.$

$$\frac{C : \mathrm{CR} \quad L : \mathrm{list}_2(\alpha)}{\begin{array}{l}\mathrm{Conjr}\alpha(L) : \alpha \\ \mathrm{Conjr}\alpha(L)^\star = (x_1)\cdots(x_n)\mathrm{listconn}(C^\star, \mathrm{map}_2((X)X^\star(x_1,\ldots,x_n), L)) \\ \mathrm{Conjr}\alpha(L)^\circ(p_1,\ldots,p_k) = \mathrm{blancform}(\mathrm{map}_2((X)(C^\circ\ X^\circ(p_1,\ldots,p_k)), L)\end{array}}$$

2.13 *Les opérations logiques en français*

Pour que la grammaire puisse exprimer les propositions mathématiques en français, il lui faut avoir des règles correspondant à chaque connecteur et quantificateur logique. La méthode traditionnelle pour exprimer la quantification est l'usage des déterminants. Pour la négation, nous avons la négation du verbe, et pour les connecteurs, la coordination.

Mais nous ne pouvons pas encore définir une traduction de tout le calcul des prédicats en français, parce que les structures mentionnées ne peuvent pas être composées d'une façon arbitraire. Par exemple, l'usage des déterminants n'atteint pas toutes les possibilités qu'offre le calcul des prédicats pour lier les variables. Nous verrons que les variables explicites offrent une méthode plus générale et souvent plus exacte pour exprimer la quantification ; mais déjà sans variables, on peut exploiter les **connecteurs progressifs** de la théorie des types pour exprimer les propositions quantifiées. L'**implication progressive** est un quantificateur universel ; la **conjonction progressive** est un quantificateur existentiel :[71]

[70] La conjonction répétée n'a pas d'usage collectif : on ne peut pas dire *la somme et de 2 et de 3*.

[71] L'idée de cette généralisation de l'implication et de la conjonction est implicite dans la sémantique constructive de Heyting (1956), qui présente $\forall$ comme une généralisation de $\supset$ et $\exists$ comme une généralisation de &. Nous avons étudié ses conséquences pour l'analyse de la langue naturelle dans Ranta (1994).

P $\rightarrow$ *si* P, P

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{P}(\Gamma) \quad B : \mathrm{P}((\Gamma,\ x : A^\star))}{\begin{array}{l}\mathrm{siP}(\Gamma, A, B) : \mathrm{P}(\Gamma) \\ \mathrm{siP}(\Gamma, A, B)^\star = (\Pi x : A^\star)B^\star \\ \mathrm{siP}(\Gamma, A, B)^\circ(m) = \mathit{si}(A^\circ(\mathrm{ind})),\ B^\circ(m)\end{array}}$$

P $\rightarrow$ P *et* P

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{P}(\Gamma) \quad B : \mathrm{P}((\Gamma,\ x : A^\star))}{\begin{array}{l}\mathrm{etP}(\Gamma, A, B) : \mathrm{P}(\Gamma) \\ \mathrm{etP}(\Gamma, A, B)^\star = (\Sigma x : A^\star)B^\star \\ \mathrm{etP}(\Gamma, A, B)^\circ(m) = A^\circ(m)\ \mathit{et}\ B^\circ(m)\end{array}}$$

Quand on se sert d'un déterminant, le domaine de quantification est exprimé par un nom commun. Les connecteurs progressifs demandent une phrase, mais tout nom commun peut être exprimé par une phrase introduite par *il existe*:[72]

P $\rightarrow$ *il existe un* CN

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{CN}(\Gamma)}{\begin{array}{l}\mathrm{existe}(\Gamma, A) : \mathrm{P}(\Gamma) \\ \mathrm{existe}(\Gamma, A)^\star = A^\star \\ \mathrm{existe}(\Gamma, A)^\circ(m) = \mathit{il\ existe\ un}(\mathrm{gen}(A))\ A^\circ(\mathrm{sg})\end{array}}$$

La négation des verbes a, logiquement, une application très bornée. Par exemple, elle ne peut pas nier une conjonction. Mais il y a une opération très effective pour nier n'importe quelle phrase:

P $\rightarrow$ *il n'est pas vrai que* P

$$\frac{\Gamma : \mathrm{cont} \quad A : \mathrm{P}(\Gamma)}{\begin{array}{l}\mathrm{pasvrai}(\Gamma, A) : \mathrm{P}(\Gamma) \\ \mathrm{pasvrai}(\Gamma, A)^\star = \sim A^\star \\ \mathrm{pasvrai}(\Gamma, A)^\circ(m) = \mathit{il\ ne}(\mathit{être}(m))\ \mathit{pas\ vrai\ que}(A^\circ(\mathrm{subj}))\end{array}}$$

Suite de l'article «Structures grammaticales dans le français mathématique» dans *Mathématiques, Informatique et Sciences humaines*, n°139, automne, 1997.

[72]Nous ne connaissons pas d'opération inverse, qui transformerait une phrase donnée en un nom commun; ce serait une opération de **nominalisation** des phrases.

BIBLIOGRAPHIE

ABEILLÉ A., *Les nouvelles syntaxes*, Paris, Armand Colin, 1993.

AJDUKIEWICZ K., « Die syntaktische Konnexität », *Studia Philosophica* 1 (1935), pp. 1–27.

BAR-HILLEL Y., « A quasi-arithmetical notation for syntactic description », *Language*, 29 (1953), pp. 47–58.

DE BRUIJN N. G., « Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem », *Indagationes Mathematicae* 34 (1972), pp. 381–392.

DE BRUIJN N. G., « The mathematical vernacular, a language for mathematics with typed sets », NEDERPELT R., éd., *Selected Papers on Automath*, pp. 865–935, Amsterdam, North-Holland, 1994.

CHAMBREUIL M., *Grammaire de Montague : langage, traduction, interprétation*, ADOSA, Clermont-Ferrand, 1989.

CHOMSKY N., *Aspects of the Theory of Syntax*, Cambridge, Ma., The M.I.T. Press, 1965.

COSCOY Y., KAHN G. et THÉRY L., *Extracting text from proofs*, Rapport de recherche n. 2459, Sophia-Antipolis, INRIA, 1995.

FREGE G., *Begriffsschrift*, Halle A/S, Louis Nebert, 1879.

GAZDAR G., KLEIN E., PULLUM G. et SAG I., *Generalized Phrase Structure Grammar*, Oxford, Basil Blackwell, 1985.

GEACH P., *Reference and Generality*, Ithaca, New York, Cornell University Press, 1962.

GENTZEN G., « Untersuchungen über das logische Schliessen », *Mathematische Zeitschrift*, 39 (1934), pp. 176–210 et 405–431.

GREVISSE M., *Le bon usage*, Treizième édition, Paris, Duculot, 1993.

HEYTING A., *Intuitionism*, Amsterdam, North-Holland, 1956.

LAMBEK J., « The mathematics of sentence structure », *American Mathematical Monthly*, 65 (1958), pp. 154–170.

LAMPORT L., *LaTeX. A Document Preparation System*, Reading, Ma., Addison-Wesley Publishing Company, 1986.

LECOMTE A., *Modèles logiques en théorie linguistique*, Synthèse de travaux présentés en vue de l'habilitation à diriger des recherches, Université de Grenoble, 1994.

MAGNUSSON L., *The Implementation of ALF — a Proof Editor Based on Martin-Löf's Monomorphic Type Theory with Explicit Substitutions*, Thèse doctorale, Department of Computing Science, University of Göteborg, 1994.

MARTIN-LÖF P., *Intuitionistic Type Theory*, Naples, Bibliopolis, 1984.

MONTAGUE R., *Formal Philosophy*, Collected papers edited by R. THOMASON, New Haven, Yale University Press, 1974.

MORRILL G., *Type Logical Grammar*, Dordrecht, Kluwer Academic Publishers, 1994.

NORDSTRÖM B., PETERSSON K. et SMITH J., *Programming in Martin-Löf's Type Theory. An Introduction*, Oxford, Clarendon Press, 1990.

PAULSON L., *ML for the Working Programmer*, Cambridge, Cambridge University Press, 1991.

VON PLATO J., «The axioms of constructive geometry», *Annals of Pure and Applied Logic*, 76 (1995), pp. 169–200.

RANTA A., *Type Theoretical Grammar*, Oxford, Oxford University Press, 1994.

RANTA A., «Syntactic categories in the language of mathematics», DYBJER P., NORDSTRÖM B. et SMITH J., éd., *Types for Proofs and Programs*, pp. 162–182, *Lecture Notes in Computer Science* 996, Heidelberg, Springer-Verlag, 1995.

RANTA A., «Context-relative syntactic categories and the formalization of mathematical text», BERARDI S. et COPPO M., éd., *Types for Proofs and Programs*, pp. 231–248, *Lecture Notes in Computer Science* 1158, Heidelberg, Springer-Verlag, 1996.

SHIEBER S., *An Introduction to Unification-Based Approaches to Grammar*, Menlo Park, Ca., CSLI, 1986.

STEEDMAN M., «Combinators and grammars», OEHRLE R., BACH E. et WHEELER D., éd., *Categorial Grammars and Natural Language Structures*, pp. 417–442, Dordrecht, D. Reidel, 1988.

TASISTRO A., *Formulation of Martin-Löf's Monomorphic Theory of Types with Explicit Substitutions*, Thèse de licence, Department of Computing Science, University of Göteborg, 1993.