



**HAL**  
open science

# Exploration de la Grammaire Catégorielle Combinatoire Multimodale

Simon Delamarre

► **To cite this version:**

Simon Delamarre. Exploration de la Grammaire Catégorielle Combinatoire Multimodale. Informatique et langage [cs.CL]. 2011. dumas-00636161

**HAL Id: dumas-00636161**

**<https://dumas.ccsd.cnrs.fr/dumas-00636161>**

Submitted on 26 Oct 2011

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

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

# Exploration de la Grammaire Catégorielle Combinatoire Multimodale

Simon Delamarre  
Master 2 de Recherche en Informatique  
TELECOM Bretagne

Encadrant : Ismaïl Biskri  
Chercheur au Département Mathématiques et Informatique Appliquées  
Université du Québec à Trois-Rivières.

13 août 2011



## Résumé

Ce rapport de stage propose l'étude d'un formalisme du Traitement Automatique des Langues (TAL), la Grammaire Catégorielle Combinatoire Multi-Modale (GCC-MM) de Jason Baldrige, extension de la Grammaire Catégorielle Combinatoire (GCC) de Mark Steedman. Un troisième formalisme, les *Categorical Type Logics* (CTL), issu du courant logique du TAL, y joue également un rôle central.

Après avoir mis en évidence (tant théoriquement qu'empiriquement) les limites inhérentes à GCC, notamment en matière de contrôle des faux positifs (surgénération), nous présentons l'extension GCC-MM et les solutions qu'elle apporte à ces problèmes. Nous détaillons les fondements formels de GCC-MM, et les liens étroits que ceux-ci établissent entre ce formalisme et les CTL, notamment via l'importation de la notion de contrôle modal. Partant de cette idée, nous entreprenons ensuite de la systématiser en dessinant les étapes d'une *méthode* générale d'utilisation conjointe des formalismes GCC-MM et CTL, visant à combiner leurs avantages respectifs. Nous tentons enfin d'appliquer cette méthode pour transposer une analyse CTL existante d'un phénomène linguistique complexe, les pronoms clitiques du français.

# Table des matières

<b>1</b>	<b>Grammaires Catégorielles Combinatoires</b>	<b>4</b>
1.1	Notions de base sur les Grammaires Catégorielles . . . . .	4
1.2	La Grammaire Catégorielle Combinatoire . . . . .	6
1.3	La Grammaire Catégorielle Combinatoire Applicative . . . . .	7
<b>2</b>	<b>Les <i>Categorical Type Logics</i></b>	<b>9</b>
2.1	Notions élémentaires sur les CTL : la base logique . . . . .	9
2.2	Notions élémentaires sur les CTL : les règles structurelles . . . . .	11
2.3	Un exemple détaillé : le traitement CTL des clitiques du français (Kraak) . . . . .	13
2.3.1	Les données linguistiques à modéliser . . . . .	13
2.3.2	La modélisation . . . . .	13
2.3.3	Analyses . . . . .	15
<b>3</b>	<b>GCC et le problème de surgénération</b>	<b>18</b>
3.1	Composition croisée : motivations linguistiques . . . . .	18
3.2	Composition croisée : surgénérations . . . . .	20
3.3	Étude du cas du français . . . . .	21
3.4	Autres cas de surgénération . . . . .	23
<b>4</b>	<b>La Grammaire Catégorielle Combinatoire Multi-Modale</b>	<b>24</b>
4.1	Règles GCC et preuves CTL . . . . .	24
4.2	Analyse de Baldrige : introduction des modalités . . . . .	25
4.3	Applications linguistiques : gain par rapport aux analyses précédentes, limites . . . . .	27
4.4	Bilan . . . . .	31
<b>5</b>	<b>Fonder les extensions de GCC-MM par CTL</b>	<b>32</b>
5.1	L'exemple du combinateur D . . . . .	32
5.2	Discussion : liens entre CTL, GCC-MM, GCCA . . . . .	35
5.3	Récapitulation des analogies entre GCC-MM et CTL et GCCA . . . . .	38
<b>6</b>	<b>Application : Conception d'un système MM-CCG pour le traitement des clitiques du français</b>	<b>39</b>
6.1	Élaboration des règles MM-CCG compilant l'analyse de Kraak . . . . .	39
6.2	Résultats . . . . .	42
6.3	Bilan et travaux futurs . . . . .	45

## Introduction

Selon une formule répandue, le but idéal vers lequel doit s'efforcer de tendre tout système visant à modéliser la syntaxe des langues naturelles est le pouvoir de reconnaître « toutes les phrases d'une langue, et seulement elles ». Cette expression, quoique nuancable, a le mérite de synthétiser deux composantes essentielles à la base du travail de conception de grammaires formelles pour cette tâche de modélisation. Reconnaître « toutes les phrases d'une langue » fait référence à la tâche d'augmenter la **couverture des grammaires**, c'est-à-dire la partie du langage dont elles sont capables de rendre compte. Bien souvent, les langues naturelles présentent une diversité de structures et une souplesse difficiles à intégrer dans des systèmes formels (libertés dans l'ordre des mots, constructions inversées, exceptions...), et le succès de leur traitement peut nécessiter de longs raffinements successifs de ces systèmes, qui représentent un véritable défi de recherche. Ces raffinements peuvent consister, par exemple, en l'introduction de nouveaux « degrés de libertés » pour gérer la variabilité inhérente aux langages naturels. Tout au long du rapport, nous nous référerons à ce premier type de problèmes (de couverture incomplète de la langue, i.e. de faux négatifs) par le terme **sous-génération**.

La fin de la phrase (« toutes les phrases d'une langue... *et seulement elles.* ») renvoie elle au symétrique de la première tâche : s'assurer que les modèles n'autorisent pas des constructions interdites par la langue modélisée, c'est-à-dire de contrôler les faux positifs induits par les modèles. On désignera ce deuxième type de problème par le terme **surgénération**. Les problèmes de surgénération peuvent parfois apparaître à première vue comme secondaires par rapport à la première catégorie ; en effet il est vrai que pour plusieurs applications pratiques (analyses de textes littéraires, d'articles journalistiques, scientifiques...) supposer que les analyses porteront essentiellement sur des phrases correctes est une hypothèse raisonnable. Cependant, dans d'autres contextes (par exemple les logiciels pédagogiques, ou les applications impliquant la génération automatique de phrases en langage naturel) il est, au contraire, tout à fait crucial de pouvoir explicitement distinguer les constructions incorrectes des valides. Ces deux premiers critères, limitation de la *sous-génération* et de la *surgénération*, doivent donc être évalués conjointement.

La complexité de la tâche réside dans le fait que les deux tâches d'éradication de la sous- et surgénération, sont éminemment intriquées. En effet, à mesure que l'on tente d'étendre la couverture d'une grammaire, on est typiquement amené à introduire de nouveaux degrés de liberté, par exemple de nouvelles façons de combiner les mots. Mais cependant, en permettant l'inclusion de ces nouvelles structures, on s'expose parallèlement au danger de surgénération, d'ouvrir la porte à des constructions indésirées. Si de telles surgénérations surviennent, on comprend qu'on cherchera à les résoudre en imposant des contraintes supplémentaires, contraintes qui, elles, induisent en retour un risque de sous-génération... Sortir de ce cercle vicieux impose de se doter d'outils formels puissants capables de contrôler l'impact des ajouts effectués à un modèle.

Puisque les buts visés concernent le Traitement *Automatique* des Langues, un troisième critère crucial est le **coût algorithmique** associé aux solutions mises en œuvre. Celui-ci constitue en quelque sorte une mesure du « réalisme » d'une solution, assurant que des implémentations effectives retourneront un résultat en un temps acceptable par rapport aux applications envisagées (ou les disqualifiant), et donc, déterminant *in fine* l'applicabilité d'une méthode pour résoudre des problèmes du « monde réel ».

Si elle résume de façon satisfaisante les deux tâches de base sous-jacentes à la conception des grammaires, on pourra cependant reprocher à la formule énoncée d'être trop simplificatrice sur certains aspects, notamment en considérant la question de l'appartenance ou non à une langue naturelle comme une question binaire, par analogie avec les langages formels. Or, dans la réalité, pour les langues naturelles, cette notion est bien plus mouvante. Tout d'abord, l'acceptation d'une phrase ou non peut être sujette à polémique. Il est par exemple souvent difficile de distinguer les situations de réelle agrammaticalité et certaines constructions théoriquement correctes mais sanctionnées par l'usage. L'existence des registres de langage est une autre limite à cette dichotomie : on pourrait poser que seules sont correctes les phrases appartenant au registre standard. Cependant ne pas pouvoir traiter les constructions familières, par exemple, peut se révéler handicapant dans un grand nombre de contextes (l'hypothèse supposant que les phrases analysées seront écrites dans un style standard est en effet souvent trop forte, notamment du fait qu'un grand nombre de textes, y compris journalistiques et littéraires -sans parler des e-mails, blogs, commentaires d'articles...-, sont susceptibles de contenir du discours rapporté dont le registre n'est pas prévisible (ainsi nombre des personnages des *Misérables* emploient vocabulaire et

constructions argotiques). Dans le cas des langues vivantes, tous ces paramètres (grammaticalité, appartenance à un registre...), sont en outre susceptibles d'évoluer avec le temps et les réformes linguistiques. Ceci nous amène à formuler un **quatrième** (et dernier) **critère** informel pour évaluer un modèle de syntaxe des langues : sa souplesse par rapport à la notion de grammaticalité (par exemple, sa capacité de rejeter ou accepter, selon le besoin, les phrases d'un certain registre, et de détecter l'appartenance au dit registre), et sa robustesse vis-à-vis d'éventuelles évolutions de la langue (typiquement, devant une phrase déclarée douteuse « ? »<sup>1</sup>, une grammaire doit idéalement mettre en évidence sa capacité à permettre aussi aisément le rejet que l'acceptation de la phrase).

Le présent rapport de stage présente les résultats d'un travail d'exploration d'un formalisme pour le traitement des langues appartenant à la famille des Grammaires Catégorielles : la Grammaire Catégorielle Combinatoire Multi-Modale (GCC-MM, [2]). Nous verrons en particulier que le modèle GCC-MM résulte d'une extension de la Grammaire Catégorielle Combinatoire (GCC, [27]) construite sur la base d'un formalisme logique, les *Categorial Type Logics* (CTL, [22]), qui joue le rôle d'un outil formel de conception (et de contrôle de l'impact) des modifications du modèle CCG, et évaluerons les bénéfices de cette méthode par rapport aux quatre critères énoncés ci-dessus.

Les deux premières parties du rapport sont consacrées à la présentation des outils formels exploités dans toute la suite, i.e. les formalismes GCC et CTL. La troisième partie s'intéresse aux phénomènes de surgénération existants au sein du formalisme GCC, en particulier ceux découlant des « règles de compositions croisées ». Après analyse attentive des causes théoriques engendrant ces phénomènes, nous présenterons la solution des restrictions de règles [27], et ses limites. Cet appel à la littérature sera complété par une étude empirique des surgénérations de GCC dans le cas du français, via la constitution d'un corpus de faux positifs pour cette langue, justifiant que leur survenance n'est en rien anecdotique. Dans une quatrième partie, nous introduirons le formalisme de la GCC-MM proprement dit. Nous mettrons en évidence comment le système de modalités qu'il introduit offre une réponse plus souple aux problèmes de surgénération, et permet de dépasser certaines des limites apparaissant dans les précédentes solutions à base de restrictions de règles (en soulignant cependant que certaines difficultés demeurent).

Dans la cinquième partie nous porterons notre attention plus particulièrement sur la *méthode* sous-jacente à la construction de GCC-MM (c'est-à-dire l'appui sur CTL pour fonder des extensions de GCC de façon contrôlée, en tirant parti des avantages respectifs des deux formalismes). Nous montrerons que cette approche peut efficacement être généralisée pour élaborer de nouvelles règles GCC et étendre ce faisant la couverture du système (nous présenterons l'exemple des règles « D ») tout en contrôlant la surgénération. La septième (et dernière) partie est une application de cette méthode. Nous y proposerons une analyse des pronoms clitiques du français dans le cadre des grammaires catégorielles, fondée sur la base d'une analyse CTL existante (de laquelle sera dérivé un système de modalités du type GCC-MM). À la lumière du bilan de cette tentative, nous discuterons des rôles respectifs de CTL et GCC-MM dans cette méthode, en tentant de définir plus précisément le cadre de leur interaction.

## 1 Grammaires Catégorielles Combinatoires

Les deux premières parties ont vocation à donner au lecteur les outils formels nécessaires à la compréhension des suivantes, plus qu'à établir un état de l'art exhaustif des formalismes utilisés. Nous commençons par la Grammaire Catégorielle Combinatoire (GCC), formalisme à la base de la Grammaire Catégorielle Combinatoire Multi-Modale, qui est le sujet principal du présent rapport.

### 1.1 Notions de base sur les Grammaires Catégorielles

Les Grammaires Catégorielles sont une famille de formalismes dédiés à la modélisation de la syntaxe des langues naturelles, issus des travaux de Ajdukiewicz [1] et Y. Bar-Hillel [5]. Ces grammaires reposent sur deux idées fondamentales :

---

1. Nous suivrons dans ce rapport la convention usuelle consistant à faire précéder les phrases à la grammaticalité incertaine par un point d'interrogation ?, et les phrases définitivement agrammaticales par une astérisque \*.

La première de ces idées est l'assignation de **types syntaxiques orientés** aux catégories du lexique (nom, verbe, adjectif, adverbe...) selon un schéma **opérateur/opérande**. Formellement, ce schéma est défini par la donnée d'un ensemble de types atomiques  $\mathcal{T}_0$  et de deux lois de composition entre les types  $\{/\backslash\}$  (*slash avant* et *arrière*) qui engendrent l'ensemble de tous les types  $\mathcal{T}$ , par la définition récursive suivante :

$$\mathcal{T} := \mathcal{T}_0 | \mathcal{T} / \mathcal{T} | \mathcal{T} \backslash \mathcal{T}$$

Typiquement, on a  $\mathcal{T}_0 = \{N, NP, S\}$ , qui dénotent respectivement les types du nom (*Noun*), du groupe nominal saturé (*Noun Phrase*) et de la phrase (*Sentence*). Certains auteurs ([27] par exemple) introduisent aussi le type *PP*, groupe prépositionnel.

Les autres types sont donc construits récursivement à partir de ces types de bases, comme des fonctions d'autres types.  $X/Y$  (resp.  $X\backslash Y$ ) dénote un type opérateur qui prend sur sa **droite** (resp. sur sa **gauche**) une opérande de type  $Y$  pour former un résultat de type  $X$ . Ces types sont associés aux mots d'une langue dans un **lexique**  $\mathcal{L}$ , qui constitue une table d'association mot-type syntaxique. Un verbe intransitif (*dormir* exemple), qui attend sur sa gauche un groupe nominal complet (par exemple *Le chat*, ou *Mafalda* : NP) pour former une phrase (syntaxiquement correcte) se verra ainsi attribuer le type  $S\backslash NP$ . Un verbe transitif (comme *manger*) prend en plus un objet sur sa droite, et se verra donc attribuer le type  $S\backslash NP/NP$ . Un déterminant comme *Le* aura le type  $NP/N$ . Un adjectif, qui (en français) qualifie le nom qui le *précède*, sera associé au type  $N\backslash N$ . En anglais, où c'est habituellement l'adjectif qui précède le nom, ce sera  $N/N$ .

La deuxième idée est la définition d'un **calcul** sur ces catégories, permettant aux opérateurs de consommer effectivement leurs opérandes et de rendre leur résultat disponible pour d'autres combinaisons. Les deux règles de ce calcul sont appelées les règles d'**application avant** ( $>$ ) et **arrière** ( $<$ ) :

$$\begin{aligned} > : X/Y \quad Y \rightarrow X \\ < : Y \quad X\backslash Y \rightarrow X \end{aligned}$$

Ces deux règles fondamentales définissent donc l'action concrète des types représentés par la notation catégorielle (i.e. la consommation d'une opérande par un opérateur pour former un résultat). Elles permettent de combiner les éléments d'une phrase pour révéler sa structure et vérifier sa bonne formation syntaxique. Donnons en guise d'exemple la dérivation de la phrase *Mafalda lit le journal* (tous les types nécessaires ont déjà été donnés dans le paragraphe précédent).

$$\frac{\frac{\text{Mafalda}}{NP} \quad \frac{\frac{\text{lit}}{S\backslash NP/NP} \quad \frac{\frac{\text{le}}{NP/N} \quad \frac{\text{journal}}{N}}{NP}}{S\backslash NP}}{S} < >$$

Le déterminant *le* consomme le nom *journal* pour former un groupe nominal saturé qui lui est consommé en temps qu'objet du verbe transitif *lire*, construisant une entité qui enfin consomme le sujet *Mafalda* pour former enfin une phrase complète (S). Ce calcul prouve ainsi la bonne correction syntaxique de notre exemple.

Un intérêt de ces grammaires est qu'on peut les lier à des grammaires « sémantiques », comme par exemple celle de Montague [21], dont le but est de donner une représentation des phrases analysées au moyen de formules du calcul des prédicats (ou d'autres systèmes plus élaborés<sup>2</sup>), en vue de traiter des problèmes d'analyse du langage plus avancés (quantification, résolution d'anaphores...).

On parle parfois d'**interface syntaxe-sémantique** pour désigner cette correspondance. Concrètement, le lien est effectué en associant à chacun des éléments du lexique une **interprétation** fonctionnelle en plus de son type syntaxique. Un formalisme couramment utilisé pour représenter ces interprétations est le  $\lambda$ -calcul<sup>3</sup>.

2. Le calcul des prédicats comporte en effet certaines dans le cadre d'une telle représentation. On peut trouver dans la littérature scientifique un certain nombre d'exemples problématiques, les fameuses *donkey sentences*, qui montrent son inadéquation pour traiter certains phénomènes de quantification qui surviennent dans les langues naturelles

3. Le  $\lambda$ -calcul est un système formel dédié à la manipulation de fonctions. Il repose sur deux concepts fondamentaux :

- L'**application** ( $u v$ ), qui permet d'appliquer une fonction  $u$  à un argument  $v$
- L'**abstraction**  $\lambda x.u$ , qui permet de créer des variables liées (ici,  $x$ ).

Les deux notions sont reliées par la réduction suivante : l'effet de  $((\lambda x.u) v)$  est de remplacer toutes les occurrences libres de  $x$  dans  $u$  par  $v$  (*substitution*).

Par exemple, à un verbe transitif comme *lire* (représenté comme un prédicat à deux arguments : son objet et son sujet), on associera la  $\lambda$ -expression suivante :  $\lambda x.\lambda y.\mathbf{lit}(x, y)$ . On construit ensuite le **calcul des structures prédictives** en associant à chaque règle combinatoire son effet sur les interprétations (un couple type syntaxique : interprétation sera noté  $[X : u]$ , on omettra les crochets dans les calculs) :

$$\begin{aligned} > : [X/Y : u_1] \quad [Y : u_2] \rightarrow [X : (u_1 u_2)] \\ < : [Y : u_1] \quad [X \setminus Y : u_2] \rightarrow [X : (u_2 u_1)] \end{aligned}$$

Réécrivons par exemple le calcul précédent en intégrant le calcul parallèle de la structure prédictive :

$$\frac{\frac{\text{Mafalda}}{\text{NP : Mafalda}} \quad \frac{\frac{\text{lit}}{\text{S} \setminus \text{NP} / \text{NP} : \lambda x.\lambda y.\mathbf{lit}(x, y)} \quad \frac{\frac{\text{le}}{\text{NP} / \text{N} : \mathbf{le}} \quad \frac{\text{journal}}{\text{N} : \mathbf{journal}}}{\text{NP : (le journal)}}}{\text{S} \setminus \text{NP} : \lambda y.\mathbf{lit}(\mathbf{le journal}, x)} < >}{\text{S : lit}(\mathbf{le journal}, \mathbf{Mafalda})} <$$

Le calcul des structures prédictives et leur exploitation n'est pas le sujet principal de ce rapport, aussi ne présenterons-nous leur construction qu'occasionnellement, lorsque des nouveautés ou difficultés particulières le rendront nécessaire.

Le système que nous avons décrit jusqu'à maintenant est connu sous le nom de **système AB** (AB pour Ajdukiewicz/Bar-Hillel). Il fait partie des formalismes dits fortement *lexicalisé*, c'est à dire que la majeure partie de l'information linguistique est contenue dans le lexique, via l'attribution de catégories syntaxiques fonctionnelles dont l'effet est défini par un petit nombre de règles fondamentales, ici, les règles d'application.

On notera qu'il est possible d'intégrer des informations linguistiques supplémentaires en ajoutant à certaines catégories de  $\mathcal{T}_0$  des labels spécifiant ces types selon la personne, genre, cas... du mot auquel ils se rapportent (par exemple  $N_{sing}, NP_{accusatif}, S_{infinitive}...$ ), et en les ajoutant à l'ensemble  $\mathcal{T}_0$  comme nouveaux types atomiques. On ne sort donc pas du système AB ce faisant. L'introduction de tels types atomiques permet de rejeter des phrases incorrectes pour mauvais accord, par exemple *\*Mafalda lit les journal*. Pour ne pas entraîner de multiplications intempestives de types et être réellement efficaces, de telles stratégies doivent être combinées à une structuration de l'ensemble de types atomiques par une relation d'héritage (voir [32] ou [20] par exemple pour une utilisation intensive de ces mécanismes).

Nous présentons maintenant le formalisme de la Grammaire Catégorielle Combinatoire, une extension du système AB développée par Steedman ([27]).

## 1.2 La Grammaire Catégorielle Combinatoire

L'apport de la Grammaire Catégorielle Combinatoire (GCC) consiste en un enrichissement du calcul sur les types, ce via l'introduction de nouvelles règles combinatoires :

- ★ Règles de **composition harmoniques** :
  - $> \mathbf{B} : [X/Y : u_1] - [Y/Z : u_2] \rightarrow X/Z : (\mathbf{B} u_1 u_2)$
  - $< \mathbf{B} : [X \setminus Y : u_1] - [Y \setminus Z : u_2] \rightarrow X \setminus Z : (\mathbf{B} u_1 u_2)$
- ★ Règles de **levée de type** :
  - $> \mathbf{T} : [X : u] \rightarrow [Y/(Y \setminus X) : (\mathbf{C}_* u)]$
  - $< \mathbf{T} : [X : u] \rightarrow [Y \setminus (Y / X) : (\mathbf{C}_* u)]$
- ★ Règles de **composition croisées** :
  - $> \mathbf{B}_\times : [X/Y : u_1] - [Y \setminus Z : u_2] \rightarrow [X \setminus Z : (\mathbf{B} u_1 u_2)]$
  - $< \mathbf{B}_\times : [Y/Z : u_1] - [X \setminus Y : u_2] \rightarrow [X/Z : (\mathbf{B} u_1 u_2)]$
- ★ Règles de substitution harmoniques et croisées

(non présentées ici car d'utilisation assez anecdotique)

Nous définirons pour l'instant  $\mathbf{B}$  et  $\mathbf{C}_*$  par les  $\lambda$ -expressions suivantes :

$$\begin{aligned} \mathbf{B} &=_{def} \lambda f.\lambda g.\lambda x.f(g(x)) \\ \mathbf{C}_* x &=_{def} \lambda x.\lambda f.f(x) \end{aligned}$$

**B** correspond à la notion de **composition** entre deux fonctions, tandis que **C\*** permet de passer d'un point de vue opérande sur une entité à un point de vue opérateur. Remarquons que **B**  $X Y Z \rightarrow X (Y Z)$  et **C\***  $X Y \rightarrow Y X$

Ces règles introduisent ainsi de nouvelles façons de combiner les catégories, au-delà des seules règles d'application, permettant de modéliser des phénomènes plus riches.

Un des intérêts notables de ce nouveau système est qu'il introduit un certain degré d'**associativité** dans le calcul. Avec le système AB, la succession des combinaisons devait suivre un ordre strict : ainsi par exemple un verbe transitif  $S \backslash NP / NP$  devait avoir consommé avant de pouvoir consommer son sujet. Avec GCC, un verbe peut consommer directement son sujet, et formant une entité qui attend l'objet sur sa droite pour former une phrase ( $S / NP$ ), le tout de façon cohérente avec le calcul des structures prédicatives.

Notre exemple de la sous-section précédente peut maintenant être analysé « de gauche à droite » :

$$\frac{\frac{\frac{\text{Mafalda}}{\text{NP : Mafalda}}}{S / (S \backslash NP) : \mathbf{C}_* \text{Mafalda}} > \mathbf{T} \quad \frac{\text{lit}}{S \backslash NP / NP : \lambda x. \lambda y. \text{lit}(x, y)}} > \mathbf{B} \quad \frac{\text{le}}{NP / N : \text{le}} > \mathbf{B} \quad \frac{\text{journal}}{N : \text{journal}} > \mathbf{B}}{\frac{\frac{\frac{S / NP : \lambda x. \text{lit}(x, \text{Mafalda})}{S / N : \mathbf{B} \lambda x. \text{lit}(x, \text{Mafalda}) \text{le}} > \mathbf{B} \quad \frac{\text{journal}}{N : \text{journal}} > \mathbf{B}}}{S : \lambda x. \text{lit}((\text{le journal}), \text{Mafalda})} > \mathbf{B}}$$

La réduction de la  $\lambda$ -expression obtenue est bien la structure précédemment calculée : **lit**((**le journal**), **Mafalda**).

Cette associativité permet aussi d'analyser certaines constructions qui étaient inaccessibles au système AB, par exemple les relatives comme *Le journal que Mafalda lit.* :

$$\frac{\frac{\frac{\text{Le}}{NP / N} \quad \frac{\text{journal}}{N}}{NP} > \quad \frac{\frac{\text{que}}{N \backslash N / (S / NP)} \quad \frac{\frac{\frac{\text{Mafalda}}{\text{NP}}}{S / (S \backslash NP)} > \mathbf{T} \quad \frac{\text{lit}}{S \backslash NP / NP}}{S / NP} > \mathbf{B}}{N \backslash N} < \quad > \mathbf{B} >$$

Cette analyse impose à **Mafalda** et **lit** de se combiner pour être consommés par le pronom relatif *que*, ce qui, comme on l'a vu n'était pas possible au sein du système AB.

Notons que les règles de composition doivent en fait être généralisées, par exemple [27] définit ainsi la forme générale de la composition harmonique :

$$> \mathbf{B}^n : [X / Y : u_1] - [Y / Z / \$_1 : u_2] \rightarrow X / Z / \$_1 : (\mathbf{B}^n u_1 u_2)$$

Où  $\alpha / \$_1$  est un schéma sur l'ensemble  $\{\alpha / X_1, \alpha / X_1 / X_2, \dots, \alpha / X_1 / \dots / X_n, \dots\}$ . On définit de même  $\alpha \backslash \$_1$  (resp.  $\alpha \$_1$ ) comme schémas sur les ensembles  $\{\alpha \backslash X_1, \alpha \backslash X_1 \backslash X_2, \dots, \alpha \backslash X_1 \backslash \dots \backslash X_n, \dots\}$  (resp.  $\{\alpha | X_1, \alpha | X_1 | X_2, \dots, \alpha | X_1 | \dots | X_n, \dots\}$  avec chaque  $|$  décrivant  $\{/, \backslash\}$ ).

En pratique, l'arité des fonctions apparaissant dans la langue étant bornée, on se restreint à des petites valeurs de  $n$ , typiquement  $n \leq 3$ , ce qui crée les combinaisons  $> \mathbf{B}^2$ ,  $> \mathbf{B}^3$  en plus de  $> \mathbf{B}$ . Ceci rend possible la composition avec des verbes à deux objets comme *donner* :  $S \backslash NP / NP / NP$ .

Concernant l'aspect algorithmique, le système AB appartient à la classe des grammaires hors contexte (résultat démontré par Bar-Hillel), et est donc analysable en temps  $O(n^3)$  (où  $n$  est la taille de la phrase entrée en mots) par l'algorithme de Cocke-Younger-Kasami. GCC n'est plus hors-contexte, cependant il existe un algorithme capable de l'analyser en  $O(n^6)$ , élaboré par [30]. Ceci classe GCC dans la classe (informelle) des langages dits **faiblement contextuels**, c'est à dire des langages contextuels pour lesquels il existe un algorithme d'analyse en temps polynomial.

### 1.3 La Grammaire Catégorielle Combinatoire Applicative

Nous utilisons également dans ce rapport un certain nombre de concepts appartenant à la Grammaire Catégorielle Combinatoire Applicative (GCCA) de JP. Desclés et I. Biskri ([7], [11]).



Le premier est l'utilisation de la **Logique Combinatoire** pour le calcul des expressions prédicatives (à la place du  $\lambda$ -calcul). La Logique Combinatoire est un formalisme issu des travaux de Schönfinkel [26] puis Curry et Feys [12], qui permet de se passer du concept de variable liée par l'introduction de **combinateurs**, objets qui être vus comme des processus élémentaires de calcul. Un combinateur est défini par une règle de  $\beta$ -réduction qui traduit son action sur une suite d'arguments. Ci dessous quelques exemples de combinateurs avec leurs règles de  $\beta$ -réduction associées :

- combinateur d'effacement  $\mathbf{K} X Y \rightarrow X$
- combinateur de substitution  $\mathbf{S} X Y Z \rightarrow X Z (Y Z)$
- combinateur d'identité  $\mathbf{I} X \rightarrow X$
- combinateur de composition  $\mathbf{B} X Y Z \rightarrow X (Y Z)$
- combinateur de changement de type  $\mathbf{C}_* X Y \rightarrow Y X$
- combinateur de duplication  $\mathbf{W} X Y \rightarrow W X Y Y$

*Note* (convention de suppression des parenthèses) :  $X Y Z =_{def} ((X Y) Z)$

Une propriété remarquable est que tous les combinateurs peuvent être engendrés à partir des seuls combinateurs  $\mathbf{S}$  et  $\mathbf{K}$  (formellement, c'est en fait une définition : les objets de la logique combinatoire sont définis comme le système engendré par  $\mathbf{S}$  et  $\mathbf{K}$ ). On a déjà rencontré (indirectement) les  $\mathbf{B} \mathbf{C}_*$  en les définissant par des  $\lambda$ -expressions. Il est possible de transformer toute  $\lambda$ -expressions en expression de la logique combinatoire sans variables liées (et réciproquement, cependant cette correspondance n'est pas bijective). Plusieurs algorithmes réalisant une telle transformation existent (voir [12]). Dans ce rapport nous utiliserons la transformation  $T[\ ]$  ci-dessous :

$$\begin{aligned}
T[(uv)] &= (T[u] T[v]) \\
T[\lambda x.x] &= \mathbf{I} \\
T[\lambda x.u] &= \mathbf{K} T[u] \text{ si } x \text{ n'est pas libre dans } u \\
T[\lambda x.\lambda y.u] &= T[\lambda x.T[\lambda y.u]] \\
T[\lambda x.(fx)] &= T[f] \text{ si } x \text{ n'a pas d'occurrence libre dans } f \text{ (}\eta\text{-réduction)} \\
T[\lambda x.(uv)] &= \mathbf{B} u T[\lambda x.v] \text{ si } x \text{ a une occurrence libre dans } v, \text{ mais pas dans } u \\
T[\lambda x.(uv)] &= \mathbf{C} T[\lambda x.u] v \text{ si } x \text{ a une occurrence libre dans } u, \text{ mais pas dans } v \\
T[\lambda x.(uv)] &= \mathbf{S} T[\lambda x.u] T[\lambda x.v] \text{ si } x \text{ a une occurrence libre dans } u \text{ et } v
\end{aligned}$$

Un intérêt de l'utilisation des combinateurs et de la suppression de variables liées est de pouvoir se libérer de l'impératif de non-duplication des noms de variables, et ainsi d'éviter d'avoir à utiliser des mécanismes complexes de renommage (ce qui est appréciable pour l'implémentation des algorithmes d'analyse). Ils sont également employés pour des analyses avancées dans le domaine de l'analyse des structures prédicatives (voir [10] par exemple), que nous ne considérons pas ici.

La deuxième idée empruntée plus spécifiquement à la GCCA (bien qu'elle soit également évoquée, dans dans un sens moins strict, dans GCC, c.f. [27], [19]) est l'adoption **stratégie d'analyse incrémentale**. Celle-ci consiste à analyser les phrases « de gauche à droite », en cherchant à combiner les mots dans l'ordre par lequel ils apparaissent dans la phrase : *((Mafalda lit) le) journal* plutôt que *(Mafalda (lit (le journal)))*.

Ce type d'analyse « de gauche à droite » est rendu possible par le degré d'associativité introduit par les règles de GCC (nous verrons cependant que dans certaines situations, celui-ci n'est pas suffisant pour mener à bien une telle stratégie).

Outre ses motivations psycholinguistiques<sup>4</sup>, une telle stratégie présente d'importants intérêts algorithmiques. Elle permet notamment, en privilégiant un sens d'analyse, d'éviter le phénomène dit de la **fausse ambiguïté** (*spurious ambiguity*). Cette « fausse ambiguïté » est une conséquence directe du degré de liberté laissé par GCC par rapport à l'ordre des combinaisons : plusieurs stratégies de combinaison peuvent mener à une même structure prédicative (voir les deux dérivations de notre exemple *Mafalda lit le journal*), et un analyseur explorant toutes les stratégies de combinaison engendrera une structure autant de fois qu'il y a de dérivations y menant, ce qui est sous-optimal.

4. Un certain nombre d'arguments de cet ordre semble démontrer assez nettement que les être humains eux-mêmes analysent les phrases lues ou entendues selon une stratégie semblable, au fur et à mesure, sans en attendre la fin. Voir par exemple le concept de *garden path* dans [19]

La possibilité d’une analyse incrémentale est en outre cruciale pour un certain nombre d’applications, par exemple la correction orthographique à la volée, ou la dictée par reconnaissance vocale (voir [16] par exemple).

Signalons également que la GCCA se distingue de GCC par la définition explicite d’un système de **métarègles**, qui contrôlent le déclenchement des règles de levée de type.

Ceci achève notre présentation des mécanismes de base de la Grammaire Catégorielle Combinatoire. Nous abordons maintenant l’introduction au deuxième formalisme employé dans ce rapport, les *Categorical Type Logics* (ou CTL).

## 2 Les *Categorical Type Logics*

Parallèlement à la conception *combinatoire* des grammaires catégorielles que nous venons de présenter, un autre courant s’est développé à partir de 1958 et les travaux de Lambek [18] adoptent une approche *déductive* au problème de l’analyse syntaxique, et offrent une solution alternative pour étendre le système AB.

Ce point de vue diffère du premier en rapprochant le concept d’analyse syntaxique de celui de *preuve* logique. Les systèmes qui en font partie reposent sur la définition d’**axiomes** qui définissent les propriétés générales du système déductif, et desquels on déduit des **théorèmes** qui sont l’équivalent des règles stipulées dans les systèmes combinatoires. Analyser une phrase revient à *prouver* qu’elle dérive un certain type.

Les *Categorical Type Logics* (CTL) sont une famille de formalismes appartenant à ce courant logique, caractérisé par l’introduction de la notion de *contrôle de ressources* basé sur la Logique Linéaire [13]. Ils visent notamment à introduire de façon contrôlée (via la notion de **modalité**) des opérations comme l’associativité ou la permutativité, afin de pouvoir modéliser un éventail plus large de phénomènes linguistiques.

### 2.1 Notions élémentaires sur les CTL : la base logique

Le système de catégories utilisées par les CTL est très semblable à celui du système AB et de GCC, mis à part le fait que les slashes peuvent être décorés par un ensemble (fini) de **modalités binaires**  $\mathcal{M}_b$  qui traduisent différents modes de composition linguistique. CTL introduit également des **modalités unaires**  $\mathcal{M}_u$  s’appliquant à une seule catégorie.

L’ensemble des types CTL est alors défini de la manière suivante :

$$\mathcal{T} := \mathcal{T}_0 | \mathcal{T} /_i \mathcal{T} | \mathcal{T} \setminus_i \mathcal{T} | \diamond_j \mathcal{T} | \square_j \mathcal{T}$$

avec  $i \in \mathcal{M}_b, j \in \mathcal{M}_u$

(Certaines versions, que nous ne considérons pas dans ce rapport, incluent un produit binaire entre les types  $\mathcal{T} \bullet_i \mathcal{T}$ ). NB : Dans tout ce rapport, nous utilisons la *notation de Steedman* pour les types fonctionnels arrières : dans l’expression  $X \setminus Y$ ,  $Y$  est l’argument et  $X$  le résultat de la fonction (contrairement à la *notation de Lambek*, plus couramment utilisée en CTL où ce type serait noté  $Y \setminus X$ ).

Comme pour le système AB et GCC, on a généralement  $\mathcal{T}_i = \{N, NP, S\}$ , éventuellement augmenté en faisant porter aux catégories des marqueurs de genre, nombre, personne, cas...

L’élément de base d’une preuve CTL est le *séquent*. Il se présente sous la forme :

$$\mathcal{S} \vdash \mathcal{F}$$

La partie gauche  $\mathcal{S}$  est un ensemble de formules prémisses (qui peuvent être soit des éléments lexicaux, soit des éléments de  $\mathcal{S}$ ) *structurées* par les différents modes de compositions. La partie droite  $\mathcal{F}$  est une formule résultante.  $\mathcal{S} \vdash \mathcal{F}$  signifie que l’on déduit la formule  $\mathcal{F}$  à partir de la structure de formules prémisses  $\mathcal{S}$ . Prenons quelques exemples. Les séquents les plus simples sont les entrées lexicales, leur côté gauche est un mot (compris en tant qu’unité graphique), et le côté droit est le type associé à ce mot. Ainsi Les séquents associés au lexique de l’exemple utilisé dans la partie précédente pourront être :

$$\begin{aligned} \text{Mafalda} &\vdash \text{NP} \\ \text{lit} &\vdash \text{S} \setminus_c \text{NP} /_c \text{NP} \end{aligned}$$

$$\begin{array}{l} \text{le} \vdash \text{NP}/_c\text{N} \\ \text{journal} \vdash \text{N} \end{array}$$

Où  $c$  est une modalité binaire quelconque, employée ici uniquement à des fins d'illustration.

L'enjeu de toute théorie du traitement de la syntaxe des langues naturelles est de modéliser comment de tels éléments lexicaux peuvent être articulés en phrases. La finalité de toute démonstration CTL est ainsi de construire des séquents plus élaborés à partir de séquents lexicaux élémentaires. Pour cela, CTL est muni d'un ensemble de **règles d'inférence** qui permettent de combiner le matériel linguistique en des structures, via différents modes de composition binaires  $\circ_i$  et unaires  $\langle \rangle_j$ , et d'effectuer des déductions sur ces structures à partir des déductions sur leurs composants.  $\text{Mafalda} \circ_c \text{lit}$  est ainsi une prémisse structurée résultant de la combinaison des prémisses  $\text{Mafalda}$  et  $\text{lit}$  par le mode de composition binaire  $\circ_c$ ,  $\langle \text{NP} \rangle_j$  est la prémisse structurée résultant de la combinaison de la prémisse NP par le mode de composition unaire  $\langle \rangle_j$ . Il importe de bien comprendre cette notion de *prémisses structurées*, fondamentale au fonctionnement du formalisme CTL.

Les règles d'inférence que nous allons maintenant présenter permettent de relier ces modes de composition des prémisses aux slash  $/_i \setminus_i$  et modalités  $\diamond_j \square_j^\downarrow$  vues précédemment (qui appartiennent au domaine des types), et d'établir le mécanisme de déduction proprement dit.

Passons rapidement sur la première de ces règles, qui est l'axiome d'identité :

$$\frac{}{A \vdash A} \text{Ax}$$

Le moyen le plus élémentaire de créer une structure binaire en CTL est d'utiliser les **règles d'élimination de slash**  $E/_i$  et  $E\setminus_i$  :

$$\frac{\Gamma \vdash A/_i B \quad \Delta \vdash B}{\Gamma \circ_i \Delta \vdash A} E/_i$$

$$\frac{\Delta \vdash B \quad \Gamma \vdash A\setminus_i B}{\Delta \circ_i \Gamma \vdash A} E\setminus_i$$

Noter la relation entre les slashes  $/_i$  apparaissant dans les types de  $\Gamma$  et  $\Delta$  et le mode de composition de l'antécédent structuré  $\Gamma \circ_i \Delta$ . Les règles d'élimination de slash sont l'analogie des règles d'application avant et arrière du système AB. La preuve du fait que  $\text{Mafalda lit le journal}$  est de type S pourra s'écrire alors :

$$\frac{\text{Mafalda} \vdash \text{NP} \quad \frac{\text{lit} \vdash \text{S}\setminus_c \text{NP}/_c \text{NP} \quad \frac{\text{le} \vdash \text{NP}/_c \text{N} \quad \text{journal} \vdash \text{N}}{\text{le} \circ_c \text{journal} \vdash \text{NP}} E/_c}{\text{lit} \circ_c (\text{le} \circ_c \text{journal}) \vdash \text{S}\setminus \text{NP}} E/_c}{\text{Mafalda} \circ_c (\text{lit} \circ_c (\text{le} \circ_c \text{journal})) \vdash \text{S}} E\setminus_c$$

L'opération inverse de l'élimination est l'**introduction de slash**, qui consiste à déduire le type d'un des constituants d'un produit structurel à partir du type total et du type de l'autre constituant :

$$\frac{\Gamma \circ_i B \vdash A}{\Gamma \vdash A/_i B} I/_i$$

$$\frac{B \circ_i \Gamma \vdash A}{\Gamma \vdash A\setminus_i B} I\setminus_i$$

En tant que système déductif, CTL est capable de combiner non seulement des éléments issus du lexique, mais aussi d'introduire et retirer des *éléments hypothétiques* comme intermédiaires de calculs dans les preuves. Dans la suite du rapport, de tels éléments seront notés par des lettres minuscules de l'alphabet latin. Ce mécanisme d'**introduction et retrait d'hypothèse** est une différence notoire avec les systèmes combinatoires comme GCC, qui ne peuvent agir que sur des éléments linguistiques effectivement présents.

À l'instar de GCC, il est possible d'établir une interface syntaxe-sémantique dans CTL via l'attribution de labels au types intervenant dans les preuves. Nous avons réécrit ci dessous les règles d'introduction et élimination de slash tenant compte de tels labels ([22]) :

$$\frac{\Gamma \vdash u : A/iB \quad \Delta \vdash x : B}{\Gamma \circ_i \Delta \vdash (ux) : A} E/i$$

$$\frac{\Delta \vdash x : B \quad \Gamma \vdash u : A \setminus_i B}{\Delta \circ_i \Gamma \vdash (ux) : A} E \setminus_i$$

$$\frac{\Gamma \circ_i x : B \vdash u : A}{\Gamma \vdash \lambda x.u : A/iB} I/i$$

$$\frac{x : B \circ_i \Gamma \vdash u : A}{\Gamma \vdash \lambda x.u : A \setminus_i B} I \setminus_i$$

On voit que les règles d'élimination correspondent à la notion d'*application* du  $\lambda$ -calcul, tandis que les règles d'introduction correspondent à la notion d'*abstraction*.

L'interaction du crochet structurel  $\langle \rangle_j$  avec le type de la structure qu'il contient ou forme est elle aussi définie au travers de règles d'introduction et d'élimination pour les modalités  $\diamond_j$  et  $\square_j^\downarrow$  :

$$\frac{\langle \Gamma \rangle_j \vdash A}{\Gamma \vdash \square_j^\downarrow A} \square_j^\downarrow I \qquad \frac{\Gamma \vdash \square_j^\downarrow A}{\langle \Gamma \rangle_j \vdash A} \square_j^\downarrow E$$

$$\frac{\Gamma \vdash A}{\langle \Gamma \rangle_j \vdash \diamond_j A} \diamond_j I \qquad \frac{\Delta \vdash \diamond_j A \quad \Gamma \{ \langle A \rangle_i \} \vdash B}{\Gamma \{ \Delta \} \vdash B} \diamond_j E$$

On supposera (hypothèse la plus courante, [15]) que ces règles sont sémantiquement inertes, c'est pourquoi nous ne précisons pas les éventuels labels dans leur énoncé.

L'ensemble des règles présentées jusqu'à maintenant constitue la **base logique** de CTL, et est présent dans tout système CTL. Il contient les lois de base qui permettent de former les structures et détermine les interactions entre leurs types et ceux de leurs constituants. Nous allons maintenant aborder le concept de *règles structurelles* qui permettent de *reconfigurer* les structures construites afin de rendre compte d'une plus grande variété de phénomènes linguistiques.

## 2.2 Notions élémentaires sur les CTL : les règles structurelles

En plus de la base logique, le formalisme CTL permet de définir un deuxième type de règles, les **règles structurelles**. Tout comme les règles ajoutées par GCC au système AB, les règles du groupe structurel permettent de disposer d'une *flexibilité* supplémentaire dans l'analyse en modifiant la structure des prémisses construites par les étapes précédentes de la preuve.

Dans la partie précédente, on a vu notamment que les règles de levée de type et de composition harmonique permettaient d'introduire un certain degré d'*associativité* dans l'analyse, autorisant une analyse incrémentale de phrases comme *Mafalda lit le journal*, ou encore le traitement des relatives comme *Le journal que Mafalda lit...*

En CTL, cette associativité est introduite explicitement par l'utilisation de deux *règles structurelles d'associativité* (*Right Association*, pour l'associativité à droite et *Left Association*, pour l'associativité à gauche) :

$$\frac{\Delta_1 \circ_c (\Delta_2 \circ_c \Delta_3) \vdash X}{(\Delta_1 \circ_c \Delta_2) \circ_c \Delta_3 \vdash X} RA$$

$$\frac{(\Delta_1 \circ_c \Delta_2) \circ_c \Delta_3 \vdash X}{\Delta_1 \circ_c (\Delta_2 \circ_c \Delta_3) \vdash X} LA$$

Un point capital est que **les règles structurelles sont sensibles aux modalités** portées par les modes de composition qui forment les structures. On voit ci-dessus, par exemple, que nous avons lié les règles RA et LA au produit structurel  $\circ_c$ . Une structure ne peut donc avoir accès à une certaine règle que si elle possède la bonne configuration *et* si les modes de composition (binaires et unaires) qui la forment portent les modalités adéquates. Les modalités jouent ainsi le rôle de *contrôle* d'accès aux règles structurelles.

Voyons comment ces règles peuvent être exploitées pour déduire le type de *Mafalda lit...* et ainsi couvrir les constructions évoquées plus haut :

$$\frac{\text{Mafalda} \vdash \text{NP} \quad \frac{\text{lit} \vdash \text{S} \setminus_c \text{NP} /_c \text{NP} \quad x \vdash \text{NP}}{\text{lit} \circ_c x \vdash \text{S} \setminus \text{NP}} /_c \text{E}}{\frac{\text{Mafalda} \circ_c (\text{lit} \circ_c x) \vdash \text{S}}{(\text{Mafalda} \circ_c \text{lit}) \circ_i x \vdash \text{S}} \text{RA}} /_c \text{I}} \setminus_c \text{E}$$

Noter l'introduction de l'hypothèse  $x$  de type NP, qui joue le rôle de l'objet absent, et permet de former la structure  $\text{Mafalda} \circ_c (\text{lit} \circ_c x)$ . L'intervention de la règle d'associativité permet ensuite de réorganiser celle-ci, de manière à former le produit  $\text{Mafalda} \circ_c \text{lit}$ , de manière à déduire son type par introduction de slash.

Tout comme les règles d'introduction et élimination des modalités unaires, on considère souvent que les règles structurelles sont sémantiquement inertes.

La grande expressivité du système de règles structurelles permet de définir une grande variété de comportements et mécanismes : associativité, permutations, propagation de traits, contraintes d'ordonnancement spéciales... Un atout majeur est la possibilité d'opérer un contrôlé fin de ces degrés de libertés via le système de modalité, ce qui permet de combattre la survenance de surgénérations.

Elles peuvent aussi être utilisées pour définir une interaction entre certaines modalités (binaires ou unaires). La sous-partie 2.3 utilise par exemple la règle structurelle suivante :

$$\frac{\Gamma\{\langle \Delta \rangle_l^8 \circ_u \langle \Theta \rangle_l^8\} \vdash A}{\Gamma\{\langle \Delta \circ_u \Theta \rangle_l^8\} \vdash A} K_u$$

Qui définit une propriété de « distributivité » (voir [22] pour une approche plus formelle de ces notions) du mode de composition unaire  $\langle \rangle_l^8$  par rapport au mode binaire  $\circ_u$ .

Le revers de la médaille à l'expressivité des règles structurelles est l'augmentation de coût algorithmique associé à l'analyse. Tout comme l'introduction des règles de GCC faisait passer cette complexité de  $O(n^3)$  (cas du système AB, hors-contexte) à  $O(n^6)$  (algorithme de [30] pour l'analyse de GCC), la complexité de l'analyse syntaxique au moyen des CTL<sup>5</sup> est directement liée à la généralité des règles structurelles utilisées.

[24] établit une caractérisation précise de cette complexité et de la puissance générative du système en fonction de la généralité des règles structurelles utilisées. Il définit une propriété de *longueur* d'une structure<sup>6</sup> et démontre que les systèmes CTL n'utilisant que des règles *non-expansives* (i.e. dont la longueur de la structure résultante est plus petite ou égale à celle de la structure prémisses) génèrent la classe des langages *contextuels* (type 1). Les systèmes CTL n'imposant pas cette restriction deviennent *Turing-complets* (type 0). On a donc la correspondance suivante :

Forme des règles structurelles	Position dans la hiérarchie de Chomsky
Pas de règles structurelles	Hors-Contexte (type 2)
Règles structurelles non-expansives	Contextuel (type 1)
Cas général	Turing-complet (type 0)

Il n'existe pas d'algorithme polynomial permettant de reconnaître la classe des langages contextuels. On constate donc que, même en se restreignant à des règles structurelles non expansives (propriété qui a été vérifiée dans la plupart des applications linguistiques connues de ces formalismes, [24], [2]), l'attractive expressivité des systèmes CTL se paye par une complexité plus grande que celle des systèmes plus restreints comme GCC. Ceci constitue l'inconvénient principal de cette famille de formalismes. Nous sommes ainsi à nouveau confronté à l'habituel dilemme, entre l'impératif d'expressivité qui permet de se départir plus facilement des problèmes de surgénération et sous-génération, et l'impératif de réalisme algorithmique.

Nous allons maintenant étudier de façon détaillée un exemple de système CTL dû à E.Kraak, qui s'attaque au problème (épineux) de la modélisation des pronoms clitiques en français.

5. Nous ne présentons pas dans ce rapport les algorithmes concrets permettant d'automatiser l'analyse syntaxique par les CTL. Ceux-ci relèvent des méthodes de démonstration automatique, notamment via l'utilisation de la théorie des **réseaux de preuves** (le lecteur intéressé pourra consulter la thèse de R. Moot, [24]).

6. Cette notion de longueur d'une structure est définie récursivement comme suit :

$$\begin{aligned} \text{longueur}(\Gamma \circ_i \Delta) &= \text{longueur}(\Gamma) + \text{longueur}(\Delta) + 2 \\ \text{longueur}(\langle \Delta \rangle_i) &= \text{longueur}(\Delta) + 1 \\ \text{longueur}(\Delta) &= 0 \end{aligned}$$

## 2.3 Un exemple détaillé : le traitement CTL des clitiques du français (Kraak)

Les pronoms clitiques du français (et de plusieurs autres langues romanes) sont sujets à un certain nombre de phénomènes éminemment complexes qu'il est difficile de capturer par une grammaire formelle. Nous présentons ici l'analyse de [15] qui parvient à rendre compte d'une partie substantielle de ces phénomènes, en exploitant les outils formels de CTL.

### 2.3.1 Les données linguistiques à modéliser

[15] commence par lister les phénomènes linguistiques impliquant les clitiques à modéliser, dont nous repreneons ici les principaux :

1. Dans la plupart des constructions (hormis les impératifs positifs), les pronoms clitiques objet apparaissent dans une position inverse à celle qu'occupent les groupes nominaux normaux, c'est-à-dire avant le verbe. Exemples : *Je mange la poire. Je la mange. \*Je mange la. \*Je la poire mange.*
2. Les clitiques forment un lien fort avec le verbe auquel ils sont associés (« à mi-chemin entre le lien syntaxique et le lien morphologique », [15]), aucun mot ne pouvant se glisser entre un clitique et son verbe, si ce n'est un autre clitique. Exemples : *Marie, souvent, lui écrivait. \*Marie lui, souvent, écrivait.* Par ailleurs, deux clitiques ne peuvent être coordonnés, ni s'appliquer à une conjonction de deux verbes : *\*Je le et la voit. \*Je le voit et entend.*
3. L'ordre dans lequel peuvent apparaître les différents clitiques obéit à des contraintes strictes. Le tableau ci-dessous résume le schéma d'ordonnement que se doivent de suivre tous les clitiques :

Position	I	II	III	IV	V	VI	VII
Type de clitique	Nominatif	ne	me, te, nous, vous, se	3eme personne accusatif	3eme personne datif	y	en

TABLE 1 – Contraintes d'ordre sur les pronoms clitiques du français

Exemples : *Je ne le lui donnerai pas. \*Je ne lui le donnerai pas. Je l'y ai vu. \*J'y l'ai vu.*

À ce tableau s'ajoutent encore des contraintes supplémentaires :

- chaque position ne peut contenir qu'un seul clitique à la fois
- Les positions III et V ne peuvent être occupées simultanément. *Il présentera Mr. X au directeur. Il le lui présentera. Il te présentera au directeur. \*Il te lui présentera.*

4. Les auxiliaires et certains verbes (verbes exprimant une notion de cause comme *faire, laisser*, verbes de perception comme *voir, entendre...*) entraînent un phénomène particulier nommé « montée des clitiques ». Il y a montée d'un clitique quand celui-ci apparaît (dans l'ordre linéaire des mots) comme associé à un auxiliaire ou à certains verbes, alors qu'il est en réalité objet du verbe introduit par cet auxiliaire ou verbe introductif. Cette « montée du clitique » est obligatoire. Exemple : *J'ai trouvé la solution. /Je l'ai trouvée.* Dans le second exemple, on voit que le clitique *la* (l') est attaché à l'auxiliaire « ai » bien qu'étant bien plus vraisemblablement objet du verbe *trouver* ici au participe passé.

Comme annoncé, d'autres verbes entraînent également cette construction (exemples adaptés de [23]) : *Je fais réparer ma voiture au garagiste. Je la lui fais réparer. \*Je fais lui la réparer*

D'autres verbes, au contraire, interdisent<sup>7</sup> la montée des clitiques, ceux-ci restent associés au verbe dont il dépendent :

### 2.3.2 La modélisation

Nous présentons maintenant conjointement la modélisation de Kraak et la manière dont celle-ci permet de rendre compte des phénomènes listés ci-dessus.

7. En français moderne. Le français classique, au contraire, requerrait la montée du clitique : *Je les veux manger.*

[15] commence par distinguer trois modes de compositions : le produit  $\circ_c$  correspond au mode normal de composition entre les unités linguistiques, et le mode  $\circ_{ca}$  (*clitic-attachment*) qui dénote le mode de composition entre un clitique et son verbe (mode qui, comme on l'a vu, établit un lien prosodique fort entre le clitique et le verbe, et possède des propriétés spéciales). Le troisième mode,  $\circ_u$  (*clause union mode*), dénote une association spéciale entre deux verbes (comme auxiliaire/participe passé, semi-auxiliaire/infinif...).

Ces propriétés seront déterminées par l'interaction de ces deux modes avec les modalités unaires que nous allons maintenant présenter. [15] utilise deux modalités fondamentales :

- $\diamond_p$  (« phrasal head ») : modalité permettant de vérifier qu'une phrase est correctement structurée par rapport à sa « tête » (qui peut être son verbe principal seul, ou un complexe verbal plus sophistiqué, par exemple un verbe et ses clitiques). Une structure  $\Delta$  enchâssée dans le modalité  $p$  ( $\langle \Delta \rangle_p$ ) est « certifiée » correctement structurée par rapport à sa tête.
- $\diamond_l$  (« lexical head ») : modalité servant à valider la structure d'un complexe verbal (notamment, l'agencement des clitiques du verbe s'il en possède). Afin de pouvoir modéliser les conditions strictes sur l'ordre des clitiques au sein d'un complexe verbal, [15] rend ces modalités paramétriques, en leur associant à chacune un entier entre 0 et 8 servant à marquer la place que doit occuper une structure marquée d'une telle modalité, en référence au tableau 3 (8 étant la place du verbe). Ce qui, formellement, revient à créer autant de modalités. Les valeurs de ce paramètre sont munies de la relation d'ordre strict usuelle<sup>8</sup> sur  $\mathbb{N}$ . Ces comportements de validation de structure sont construits au moyen des règles structurelles suivantes :

Règles structurelles	Idées sous-jacentes
$\frac{\Gamma\{\langle \Delta \rangle_p^x \circ_c \Theta\} \vdash A}{\Gamma\{\langle \Delta \circ_c \Theta \rangle_p^x\} \vdash A} \text{ K1}$ $\frac{\Gamma\{\Delta \circ_c \langle \Theta \rangle_p^x\} \vdash A}{\Gamma\{\langle \Delta \circ_c \Theta \rangle_p^x\} \vdash A} \text{ K2}$	Un fragment de phrase, correctement structurée par rapport à une tête, combiné à un autre fragment via le mode de composition normal forme un fragment également bien structuré
$\frac{\Gamma\{\langle \Delta \rangle_l^y \circ_{ca} \langle \Theta \rangle_l^z\} \vdash A}{\Gamma\{\langle \Delta \circ_{ca} \Theta \rangle_l^x\} \vdash A} \text{ K}$ <p><math>x \preceq y \prec z</math></p>	<ul style="list-style-type: none"> <li>. Un complexe clitique-verbe se forme par le mode de composition <math>\circ_{ca}</math> (qui induit un lien « plus fort » que <math>\circ_c</math>). Il est bien formé si chacun des éléments qui le composent (récursivement) est bien formé.</li> <li>. Les différents composants d'un complexe verbal doivent respecter l'ordre strict défini par leur paramètre (i.e. l'ordre imposé par la langue).</li> </ul> <p>Noter que la condition « chaque place ne peut être occupée que par un seul clitique » est modélisé<sup>9</sup> par l'irréflexivité de <math>\prec</math></p>
$\frac{\Gamma\{\langle \Delta \rangle_l^8\} \vdash A}{\Gamma\{\langle \Delta \rangle_l^0\} \vdash A} \emptyset$	Il est clair que la règle précédente permet de dériver la modalité $\langle \rangle_l^i$ de toute structure verbale avec clitiques bien formée (puisque $0 \prec x$ pour tout $x$ ), et seulement d'elles. La règle $\emptyset$ permet de faire de même pour les verbes sans clitiques, conférant à $\langle \rangle_l^i$ la qualité de test général de bonne formation de tout complexe verbal.
$\frac{\Gamma\{\langle \Delta \rangle_l^x\} \vdash A}{\Gamma\{\langle \Delta \rangle_p^x\} \vdash A} I$	Un complexe verbal bien formé est à fortiori un fragment de phrase bien structuré par rapport à une tête (laquelle coïncide avec lui-même).

8. Pour gérer la condition « les places III et V ne peuvent être remplies simultanément. » -pas plus que III et IV-, [15] introduit en fait une 10ème valeur, 3', correspondant aux accusatifs première et deuxième personne, et reliée aux autres valeurs par les deux relations  $2 \prec 3' \prec 6$  (ainsi  $\prec$  n'est pas totale)

9. quoi que  $\prec$  ne fasse pas directement partie du modèle, au sens strict elle n'est ici qu'un moyen externe d'énumérer des règles structurelles.

$\frac{\Gamma\{\langle\Delta\rangle_l^s \circ_u \langle\Theta\rangle_l^s\} \vdash A}{\Gamma\{\langle\Delta \circ_u \Theta\rangle_l^s\} \vdash A} K_u$	<p>Le mode de composition <i>ou</i> permet de combiner deux verbes (auxiliaire/participe passé...) d'une façon spéciale entraînant la montée des clitiques. En effet, l'emploi de la modalité 8), interdit à un clitique d'intervenir entre ces deux verbes.</p>
$\frac{\Gamma\{\Delta \circ_i [\Theta \circ_c \Lambda]\} \vdash A}{\Gamma\{\Delta \circ_i \Theta\} \circ_c \Lambda \vdash A} MA$ <p><math>i \in \{u, ca\}</math></p>	<p>Associativité à gauche des modes <i>ca</i>, <i>u</i> par rapport au mode de composition normal <i>c</i> ([15], comme souvent en CTL cf. [22], [22], restreint l'associativité de façon à ce que le parenthésage reflète la « structure » de la phrase, ici le complexe verbal par opposition au reste de la phrase).</p>

[15], enfin, pour généraliser l'action d'un clitique sur le verbe qu'il consomme (entre autres, traduire le fait qu'un clitique comme *la* peut prendre des arguments du type  $S/NP_{acc}$  -ex : *Je la mange*- mais aussi  $S/NP_{acc}/NP_{dat}$  -*Je la donne ... (à quelqu'un)*, cas plus complexe où il faut extraire l'accusatif de sa position médiane entre le verbe et le datif-), [15] introduit un « opérateur d'extraction » noté  $\uparrow$ . Cet opérateur généralise les opérateurs du système de base ( $/$  et  $\backslash$ ) en permettant de consommer leur argument n'importe où dans une structure, ce que stipule sa règle d'introduction ([15]) :

$$\frac{\dots x \vdash B \dots}{\vdots} \frac{\Gamma\{\Delta \circ_i x\} \vdash A}{\Gamma\{\Delta\} \vdash A \uparrow_i B} I\uparrow_i$$

On voit que  $\uparrow_i$  permet de retirer un élément  $x$  où qu'il soit dans une structure (et plus seulement à sa périphérie comme  $/$  et  $\backslash$ ). L'interprétation intuitive du type  $A \uparrow_i B$  peut être formulée ainsi : un « A » auquel il manque un « B » « quelque part ».

[22], comme le signale [15], montre qu'en fait ce comportement est modélisable directement dans le cadre multi-modal à l'aide de règles structurelles adéquates, et donc qu'on peut se passer de cet « opérateur d'extraction ». Cependant, pour notre part, nous l'utiliserons nous aussi dans cette partie, afin de rendre les preuves plus concises (pour la modélisation de [22], voir partie 5).

### 2.3.3 Analyses

Voyons maintenant comment ces définitions permettent de couvrir les phénomènes inventoriés au début de cette sous-partie. Nous utiliserons l'abréviation  $SV =_{def} S \backslash NP$ .

Le paramètre de la modalité  $\langle x \rangle_l$  permet de gérer efficacement les contraintes relatives à l'ordre des critiques, en écartant toutes les surgénérations. Comparer les preuves ci-dessous de *Je lui en donnerai* et *\*J'en lui donnerai*.

Exemple : *Je lui en donnerai*

$$\frac{\frac{\text{donnerai} \vdash \square_l^{\downarrow s}(SV/cNP_{dat}/caNP_{acc})}{\langle \text{donnerai} \rangle_l^s \vdash SV/cNP_{dat}/caNP_{acc}} E \square_l^{\downarrow s} \quad x_a \vdash NP_{acc}}{\langle \text{donnerai} \rangle_l^s \circ_c x_a \vdash SV/cNP_{dat}} E/c \quad x_d \vdash NP_{dat}} E/c$$

$$\frac{\langle \langle \text{donnerai} \rangle_l^s \circ_c x_a \rangle_c \circ_c x_d \vdash SV}{\langle \langle \text{donnerai} \rangle_l^s \circ_c x_a \rangle_c \circ_c x_d \vdash SV} E/c$$

(NB : cette partie de la preuve, relativement triviale, est commune aux deux phrases)



$$\begin{array}{c}
\vdots \\
\frac{\text{en} \vdash \square_l^{\downarrow 7} \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc})}{\langle \text{en} \rangle_l^7 \vdash \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc})} \text{E}\square_l^{\downarrow 7} \quad \frac{\langle \langle \text{donnerai} \rangle_l^8 \circ_c x_a \rangle \circ_c x_d \vdash \text{SV}}{\langle \text{donnerai} \rangle_l^8 \circ_c x_d \vdash \text{SV}\uparrow_c \text{NP}_{acc}} \text{I}\uparrow_c}{\frac{\text{lui} \vdash \square_l^{\downarrow 5} \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{dat})}{\langle \text{lui} \rangle_l^5 \vdash \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{dat})} \text{E}\square_l^{\downarrow 5} \quad \frac{\langle \text{en} \rangle_l^7 \circ_{ca} (\langle \text{donnerai} \rangle_l^8 \circ_c x_d) \vdash \text{SV}}{\langle \text{en} \rangle_l^7 \circ_{ca} \langle \text{donnerai} \rangle_l^8 \vdash \text{SV}\uparrow_c \text{NP}_{dat}} \text{I}\uparrow_c} \text{E}/_c \\
\frac{\langle \text{lui} \rangle_l^5 \circ_{ca} (\langle \text{en} \rangle_l^7 \circ_{ca} \langle \text{donnerai} \rangle_l^8) \vdash \text{SV}}{\langle \text{lui} \rangle_l^5 \circ_{ca} \langle \text{en} \circ_{ca} \text{donnerai} \rangle_l^7 \vdash \text{SV}} \mathbf{K} \\
\frac{\langle \text{lui} \circ_{ca} (\text{en} \circ_{ca} \text{donnerai}) \rangle_l^0 \vdash \text{SV}}{\text{lui} \circ_{ca} (\text{en} \circ_{ca} \text{donnerai}) \vdash \square_l^{\downarrow 0} \text{SV}} \mathbf{I}\square_l^{\downarrow 0}
\end{array}$$

On constate que le complexe verbal *lui en donnerai* dérive  $\square_l^{\downarrow 0} \text{SV}$ , donc que celui-ci est bien formé. Par suite, les étapes impliquant la consommation du sujet *Je* étant triviales, la phrase est validée comme bien formée. Les deux applications de la règle K sont les deux étapes cruciales du calcul, qui n'a pu se poursuivre que parce que l'ordre des constituants formant la structure de la prémisse respectait l'ordre défini sur le paramètre de la modalité  $\langle x \rangle_l$  ( $5 \prec 7 \prec 8$ ). Pour la même raison, le fait que  $7 \succ 5 \prec 8$  permet de bloquer l'ordre incorrect *\*J'en lui donnerai*.

$$\begin{array}{c}
\vdots \\
\frac{\text{lui} \vdash \square_l^{\downarrow 5} \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{dat})}{\langle \text{lui} \rangle_l^5 \vdash \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{dat})} \text{E}\square_l^{\downarrow 5} \quad \frac{\langle \langle \text{donnerai} \rangle_l^8 \circ_c x_a \rangle \circ_c x_d \vdash \text{SV}}{\langle \text{donnerai} \rangle_l^8 \circ_c x_a \vdash \text{SV}\uparrow_c \text{NP}_{dat}} \text{I}\uparrow_c}{\frac{\text{en} \vdash \square_l^{\downarrow 7} \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc})}{\langle \text{en} \rangle_l^7 \vdash \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc})} \text{E}\square_l^{\downarrow 7} \quad \frac{\langle \text{lui} \rangle_l^5 \circ_{ca} (\langle \text{donnerai} \rangle_l^8 \circ_c x_a) \vdash \text{SV}}{\langle \text{lui} \rangle_l^5 \circ_{ca} \langle \text{donnerai} \rangle_l^8 \vdash \text{SV}\uparrow_c \text{NP}_{acc}} \text{I}\uparrow_c} \text{E}/_c \\
\frac{\langle \text{en} \rangle_l^7 \circ_{ca} (\langle \text{lui} \rangle_l^5 \circ_{ca} \langle \text{donnerai} \rangle_l^8) \vdash \text{SV}}{\langle \text{en} \rangle_l^7 \circ_{ca} \langle \text{lui} \circ_{ca} \text{donnerai} \rangle_l^5 \vdash \text{SV}} \mathbf{K} \\
\frac{\langle \text{en} \rangle_l^7 \circ_{ca} \langle \text{lui} \circ_{ca} \text{donnerai} \rangle_l^5 \vdash \text{SV}}{\text{***}} \mathbf{K}
\end{array}$$

Bien sûr, cette stratégie est générale et les autres cas d'ordonnement des clitiques (*Je les lui donne/\*Je lui les donne, Il ne s'y plaît pas./\*Il n'y se plaît pas, Je ne les lui y donnerai pas/\*J'y ne lui les donnerai pas...*) seraient automatiquement traités, aussi complexes soient-ils, avec exactement les mêmes outils, c'est à dire l'intervention de la règle K.

Étudions maintenant la question de la montée des clitiques. La mode de composition  $\circ_u$  permet d'empêcher un clitique d'intervenir entre un auxiliaire et un participe (par exemple), forçant le phénomène de montée.

Exemple . *\*J'ai la trouvée* (pour : *Je l'ai trouvée*).

$$\begin{array}{c}
\frac{\text{trouvée} \vdash \square_l^{\downarrow 8} \text{SV}_i/_c \text{NP}_{acc}}{\langle \text{trouvée} \rangle_l^8 \vdash \text{SV}_{pp}/_c \text{NP}_{acc}} \text{E}\square_l^{\downarrow 8} \quad \frac{x_a \vdash \text{NP}_{acc}}{\langle \text{trouvée} \rangle_l^8 \circ_c x_a \vdash \text{SV}_{pp}} \text{I}\uparrow_c}{\frac{\text{la} \vdash \square_l^{\downarrow 4} (\text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc}))}{\langle \text{la} \rangle_l^4 \vdash \text{SV}/_{ca}(\text{SV}\uparrow_c \text{NP}_{acc})} \text{E}\square_l^{\downarrow 4} \quad \frac{\langle \text{trouvée} \rangle_l^8 \circ_c x_a \vdash \text{SV}_{pp}}{\langle \text{trouvée} \rangle_l^8 \vdash \text{SV}_{pp}\uparrow_c \text{NP}_{acc}} \text{I}\uparrow_c} \text{E}/_c \\
\frac{\text{ai} \vdash \square_l^{\downarrow 8} (\text{SV}/_u \text{SV}_{pp})}{\langle \text{ai} \rangle_l^8 \vdash \text{SV}/_u \text{SV}_{pp}} \text{E}\square_l^{\downarrow 8} \quad \frac{\langle \text{la} \rangle_l^4 \circ_{ca} \langle \text{trouvée} \rangle_l^8 \vdash \text{SV}_{pp}}{\langle \text{ai} \rangle_l^8 \circ_u (\langle \text{la} \rangle_l^4 \circ_{ca} \langle \text{trouvée} \rangle_l^8) \vdash \text{SV}} \text{K} \\
\frac{\langle \text{ai} \rangle_l^8 \circ_u (\langle \text{la} \rangle_l^4 \circ_{ca} \langle \text{trouvée} \rangle_l^8) \vdash \text{SV}}{\langle \text{ai} \rangle_l^8 \circ_u \langle \text{la} \circ_{ca} \text{trouvée} \rangle_l^4 \vdash \text{SV}} \mathbf{K} \\
\frac{\langle \text{ai} \rangle_l^8 \circ_u \langle \text{la} \circ_{ca} \text{trouvée} \rangle_l^4 \vdash \text{SV}}{\text{***}} \mathbf{K}
\end{array}$$

La règle  $K_u$  permet d'obtenir le comportement souhaité, en faisant « sortir » les clitiques du complexe auxiliaire-participe. Ainsi la dérivation ci-dessus est contrainte de s'arrêter, car, du fait de l'inapplicabilité de cette règle à la structure auxiliaire-clitique-participe (l'adjonction d'un clitique faisant baisser le niveau du paramètre de la modalité  $\langle x \rangle_l$ ), il devient impossible de regrouper les crochets structurels des constituants en un crochet global.

*Je l'ai trouvée* produit en revanche une dérivation correcte :

$$\begin{array}{c}
\vdots \\
\frac{\langle l' \rangle_l^4 \circ_{ca} (\langle ai \rangle_l^8 \circ_u \langle trouvée \rangle_l^8) \vdash SV}{\langle l' \rangle_l^4 \circ_{ca} (\langle ai \circ_u trouvée \rangle_l^8) \vdash SV} \frac{E/u}{K_u} \\
\frac{\langle l' \rangle_l^4 \circ_{ca} (\langle ai \circ_u trouvée \rangle_l^8) \vdash SV}{\langle l' \circ_{ca} (ai \circ_u trouvée) \rangle_l^0 \vdash SV} K \\
\frac{\langle l' \circ_{ca} (ai \circ_u trouvée) \rangle_l^0 \vdash SV}{l' \circ_{ca} (ai \circ_u trouvée) \vdash \square_l^{\downarrow 0} SV} I \square_l^{\downarrow 0}
\end{array}$$

Les verbes comme *faire*, pour lesquels la montée des clitiques de leur infinitif est obligatoire, portent également un slash de type  $u$ . Le type de *faire* sera ainsi :  $\square_l^{\downarrow 8}(SV/_c NP_{dat}/_u SV_{inf})$ ; ce verbe prend donc en argument un verbe infinitif  $-/_u SV_{inf}-$  pour lequel la montée des clitiques est obligatoire, puis un argument de type NP datif (e.g. *Je fais réparer ma voiture au garagiste. / Je la lui fais réparer*). NB : cette partie se limite à la part purement syntaxique de l'analyse, sans considérer le calcul parallèle des structures prédicatives (lequel, soit dit en passant, ne pose pas de problèmes particuliers, y compris dans les cas de montée des clitiques). La dernière partie de ce rapport reprendra certains de ces calculs, avec cette fois, le calcul explicite des structures prédicatives associées.

Les verbes qui au contraire interdisent la montée des clitiques se voient attribuer le type  $\square_l^{\downarrow 8}(SV/_c \diamond \square_p^{\downarrow 0} SV_i)$ . Sans entrer dans les détails formels (qui nécessiteraient des développements supplémentaires sur la modélisation de l'opérateur $\uparrow$ ), le  $\diamond$  marquant l'argument de *voir* permet d'imposer une contrainte de domaine (on trouve souvent la terminologie *contraintes d'îlots*) sur le verbe infinitif, interdisant l'extraction à distance de son argument NP. Formellement parlant,  $\diamond$  induit un crochet structurel  $\langle \rangle$  qui, n'étant associé à aucune règle structurelle de distribution (telle K1, K2, K...) isole les composants de la structure qu'il contient (en particulier, si cette structure contient une hypothèse, celle-ci ne pourra pas en être extraite, ce qui empêchera un clitique d'atteindre l'argument NP d'un verbe infinitif introduit par *voir*<sup>10</sup>).

L'impossibilité de coordonner deux clitiques découle directement de la différence de « force » entre  $\circ_c$  et  $\circ_{ca}$  (modélisée par les règles I, K, K1 et K2). On attribue ainsi à *et* le type  $\square_l^{\downarrow x} X \setminus_c X/_c X$  : le lien de la coordination est défini de type  $\circ_c$ , plus faible que lien verbe-clitique, ce qui empêche les clitiques d'être coordonnés (car le complexe [clitique1-et-clitique2] porte la modalité  $p$  et ne peut donc se combiner ensuite au verbe via  $\circ_{ca}$ , voir ci-dessous). Ainsi, *Je mange l'abricot et la poire* est valide, mais pas *Je le et la mange..* De même pour un clitique et un complexe [verbe1-et-verbe2] (*Je le vois et l'entends/\*Je le vois et entends*).

$$\frac{\frac{\langle le \circ_c (et \circ_c la) \rangle_p^4 \vdash SV/_c (SV \uparrow_c NP_{acc}) \quad \langle mange \rangle_l^8 \vdash SV/_c NP_{acc}}{\langle le \circ_c (et \circ_c la) \rangle_p^4 \circ_{ca} \langle mange \rangle_l^8 \vdash SV} /_{ca}}{***} \mathcal{K}$$

$$\frac{\langle la \rangle_l^4 \vdash SV/_c (SV \uparrow_c NP_{acc}) \quad \langle vois \circ_c (et \circ_c entends) \rangle_p^8 \vdash SV/_c NP_{acc}}{\langle la \rangle_l^4 \circ_{ca} \langle vois \circ_c (et \circ_c entends) \rangle_p^8 \vdash SV} /_{ca}}{***} \mathcal{K}$$

Le modèle de [15] pourrait être raffiné dans de nombreuses directions (par exemple, en intégrant les effets du mode impératif sur le positionnement des clitiques, un traitement rigoureux de la particule *pas...*), cependant il réalise déjà une couverture substantielle d'un certain nombre de phénomènes complexes associés aux clitiques du français, et ce sans induire de surgénérations intempestives, du fait des mécanismes précis de gestion des ressources linguistiques dont dispose CTL. On a constaté en outre une certaine flexibilité du formalisme qui, de part les degrés de libertés dont il dispose, notamment dans l'attribution des modalités unaires et binaires, permet d'isoler un phénomène précis et de l'interdire ou l'autoriser au besoin (noter par exemple la facilité avec laquelle il est possible de passer de la syntaxe moderne à la classique et réciproquement). Ceci augure d'une bonne réutilisabilité du système, pour modéliser les constructions clitiques d'autres langues romanes par exemple.

CTL constitue donc un système attractif et expressif, qui permet de modéliser de façon concise (sans multiplication exponentielle de types) et robuste (disposant de beaucoup d'outils permettant d'éradiquer les surgénérations) des phénomènes linguistiques complexes. Le revers de la médaille, bien sûr, réside dans les

10. Si l'on désire inclure la syntaxe du français classique, il suffit d'attribuer à *vouloir* le type  $\square_l^{\downarrow 8}(SV/_u \square_p^{\downarrow 0} SV_i)$ , dans lequel on a enlevé le  $\diamond$  (qui interdisait la montée du clitique) et donné la modalité  $u$  au slash de *vouloir* (qui force la montée des clitiques).

moyens algorithmiques lourds et sophistiqués (démonstration automatique de théorèmes, réseaux de preuves) que ce formalisme nécessite pour être implémenté effectivement. C'est par cette limite que l'on « paie » l'attractive expressivité du système. Or, dans un grand nombre d'applications, voire dans la majorité, le critère de la rapidité d'exécution (et de conception de l'analyseur) est de première importance. Bien souvent, on préférera sacrifier certaines constructions complexes, admettre quelques surgénérations, au profit d'un temps de réponse viable et d'une plus grande facilité de conception (de fait, la plupart des applications actuelles liées au TAL reposent en grande partie sur des considérations statistiques, réalisant une modélisation plutôt grossière de la langue, mais néanmoins fonctionnelle dans un nombre non négligeable de contextes : classification, résumé automatique de contenus, « traduction » automatique, reconnaissance vocale...). Impératifs qui donc tendent à disqualifier CTL (du moins, dans l'état actuel de l'art ) comme formalisme autonome de Traitement du Langage... Nous verrons cependant dans les dernières parties de ce rapport comment sa combinaison avec d'autres formalismes (ici, GCC) peut se révéler éminemment féconde.

Les Grammaires Catégorielles Combinatoires, quant à elles, reposent sur un système formel moins puissant (en terme de puissance générative) mais aussi plus léger (un petit ensemble de règles simples, impliquant des mécanismes moins sophistiqués), qui la rendent plus opérationnelle (elle est analysable en temps polynomial, [29]). Nous allons voir dans la partie suivante que ces propriétés algorithmiques attractives, inévitablement, ont elles aussi un prix : l'expressivité moindre du système rend les problèmes de surgénérations beaucoup plus difficiles à maîtriser. En outre, la conception d'extensions des systèmes existants pour augmenter leur couverture peut se révéler fastidieuse, justement parce que l'impact d'une modification y est difficilement prévisible sans un contrôle empirique exhaustif. Nous nous proposons de mesurer l'étendue de ces phénomènes par une étude attentive des causes de leur survenance, en tentant de démêler les divers conflits de restrictions et d'unifications qui découlent du nombre restreint de « degrés de libertés » du système.

### 3 GCC et le problème de surgénération

Dans cette partie, nous nous intéressons aux règles de composition croisées de CCG, en détaillant d'une part les phénomènes linguistiques qui motivent leur introduction, et d'autre part les surgénérations que leur utilisation non restreinte peuvent engendrer. Nous verrons comment les restrictions de règles de Steedman ([27]) tentent d'apporter une solution à ce problème (et nous soulignerons les limites rencontrées par cette méthode). Afin de mettre en évidence le fait que ce problème n'est pas circonscrit au seul cas de la langue anglaise, nous établirons ensuite un corpus de surgénérations découlant des règles de composition croisées pour le cas du français, constituant ce faisant un argument empirique justifiant la nécessité de mettre en place des moyens de contrôle de ces phénomènes.

#### 3.1 Composition croisée : motivations linguistiques

Commençons par rappeler l'énoncé des deux règles de composition croisées qu'utilise le modèle GCC :

**Règles de composition croisées :**

$$\text{>B}_\times : [X/Y : u_1] - [Y/Z : u_2] \rightarrow [X/Z : (\mathbf{B} u_1 u_2)]$$

$$\text{<B}_\times : [Y/Z : u_1] - [X/Y : u_2] \rightarrow [X/Z : (\mathbf{B} u_1 u_2)]$$

Ces règles sont nommées *croisées* car le sens des slash des deux catégories fonctionnelles figurant dans les prémisses ont des directions opposées (contrairement aux règles dites *harmoniques* où ces directions sont identiques). Ces règles introduisent un nouveau degré de liberté, en ce sens qu'elle introduisent une nouvelle façon de combiner deux catégories. Nous allons voir que ce degré de liberté introduit une modification de « l'ordre des mots canonique » défini par les slashes des catégories.

En effet, commençons par remarquer que deux catégories figurant dans les prémisses ( $Y/Z$  et  $X/Y$ ) peuvent être reliées de façon canonique<sup>11</sup> par les dérivations suivantes :

Pour la règle  $\text{>B}_\times$  :

---

11. Par canonique, nous entendons ici sans l'introduction éventuelle de nouvelles catégories autres que celles qui figurent en position d'argument dans les catégories prémisses.

$$\frac{\frac{u_1}{X/Y} \quad \frac{\frac{u_2}{Z} \quad \frac{u_3}{Y \setminus Z}}{Y}}{X}$$

Pour la règle  $\langle \mathbf{B}_\times$ , de façon similaire :

$$\frac{\frac{\frac{u_1}{Y/Z} \quad \frac{u_2}{Z}}{Y} \quad \frac{u_3}{X \setminus Y}}{X}$$

On remarque donc que les catégories (associées à la règle d'application) produisent l'ordre  $u_1, u_2, u_3$  (en associant aux unités linguistiques fictives  $u_1, u_2$  et  $u_3$  leurs types respectifs pour chacune des deux règles), l'expression totale ayant le type syntaxique X. Il est aisé de voir que les règles croisées ont pour conséquence d'autoriser les variations  $u_2, u_1, u_3$  (pour  $\langle \mathbf{B}_\times$ ) et  $u_1, u_3, u_2$  (pour  $\langle \mathbf{B}_\times$ ).

Donnons par exemple la preuve pour  $\langle \mathbf{B}_\times$  :

$$\frac{\frac{\frac{u_1}{Y/Z} \quad \frac{u_3}{X \setminus Y}}{X/Z} \quad \frac{u_2}{Z}}{X} \langle \mathbf{B}_\times$$

L'intervention de  $\langle \mathbf{B}_\times$  permet donc bien de dériver la variante  $u_1.u_3.u_2$ , dans lequel les unités  $u_2$  et  $u_3$  ont permuté (pour donner le même type syntaxique résultant).

L'exploitation de ces degrés de liberté supplémentaires est requis par un certain nombre de phénomènes linguistiques, dont voici trois exemples (parmi beaucoup d'autres) :

– **Place de l'adverbe en français**

Ici le degré de liberté apporté par  $\langle \mathbf{B}_\times$  est exploité pour modéliser le fait que les adverbes peuvent permuter avec le complément d'objet du verbe. Comparer les dérivations des deux variantes : *Susanita aime Felipe secrètement* et *Susanita aime secrètement Felipe*.

1) L'ordre par défaut défini par les directions des slashes des catégories :

$$\frac{\frac{\frac{\frac{aime}{S \setminus NP / NP} \quad \frac{Felipe}{NP}}{S \setminus NP} \quad \frac{\frac{secrètement}{(S \setminus NP) \setminus (S \setminus NP)}}{S \setminus NP}}{NP} \quad \frac{Susanita}{NP}}{S} <$$

2) La variante permise par la règle de composition croisée arrière :

$$\frac{\frac{\frac{Susanita}{NP} \quad \frac{\frac{\frac{aime}{S \setminus NP / NP} \quad \frac{secrètement}{(S \setminus NP) \setminus (S \setminus NP)}}{S \setminus NP / NP} \quad \frac{Felipe}{NP}}{S \setminus NP}}{S} >$$

– **Constructions relatives et extraction d'objets indirects (anglais).**

On attribue généralement aux verbes ditransitifs (i.e. verbes à deux objets) anglais le type  $S/NP_{acc}/NP_{dat}$ , ce qui définit l'ordre correct des objets : le datif (objet indirect) devant obligatoirement précéder l'accusatif (objet direct), comme par exemple dans la phrase (extraite de [27]) *I give a policeman a flower* (Je donne une fleur à un policier). Cependant dans le cas de constructions relatives, comme *the policeman who I give a flower* (Le policier à qui je donne une fleur), il est nécessaire que l'accusatif puisse « permuter » par rapport à la place canonique de l'objet indirect extrait, et donc d'apparaître immédiatement après le verbe, ce qui est possible via le degré de liberté de la règle de composition croisée, comme on peut

le voir dans la preuve ci-dessous (pour simplifier, on ne montrera que les étapes les plus importantes, et on supposera également que l'on est capable de résoudre les ambiguïtés relatives aux cas des unités linguistiques en présence, et de conclure notamment que *flower* ne peut être au datif dans ce contexte).

$$\begin{array}{c}
 \frac{\text{policeman}}{N} \quad \frac{\text{who}}{N \setminus N / (S / NP_{dat})} \quad \frac{\frac{\frac{I}{NP}}{S / (S \setminus NP)} \quad T > \quad \frac{\text{give}}{S \setminus NP / NP_{acc} / NP_{dat}}}{S / NP_{acc} / NP_{dat}} > B \quad \frac{\text{a flower}}{NP_{acc}} \quad T < \\
 \frac{\text{S} / NP_{dat}}{S \setminus (S / NP_{acc})} < B_{\times} \\
 \frac{\text{N}}{N} \quad \frac{\text{N} \setminus N}{N \setminus N} >
 \end{array}$$

L'expression *policeman I give a flower* est correctement analysée comme possédant le type syntaxique N, grâce à la possibilité offerte par  $<B_{\times}$  de combiner directement *give* et *a flower*.

- **Coordination et constructions elliptiques.** Les langues à ordre SVO, telles le français et l'anglais, peuvent dans certaines constructions elliptiques accepter des constituants d'ordre SO, comme dans la phrase : *[Mafalda suit la politique]<sub>SVO</sub>*, et *[Susanita les potins.]<sub>SO</sub>*

L'enjeu est de combiner les groupes nominaux (type NP) *Susanita* et *les potins* de façon à créer un opérateur cherchant un verbe transitif sur sa gauche -ici, le verbe *suivre-* (nous ne présenterons pas ici ces calculs, assez techniques, qui impliquent de réaliser une distribution du verbe *suit*, le lecteur pourra se référer à [7] ou [9] pour le détail et une étude exhaustive des phénomènes de coordination en français). Une fois encore, ce constituant d'ordre SO peut être engendré par les règles croisées (la règle de composition croisée avant, cette fois), de la façon suivante :

$$\frac{\frac{\text{Susanita}}{NP} > T \quad \frac{\text{les potins}}{NP} < T}{S \setminus (S \setminus NP) \setminus (S \setminus NP / NP)} > B_{\times}$$

Cependant, ce degré de liberté est à double tranchant : en effet, on ne dispose d'aucun moyen de contrôler que les variantes de l'ordre des mots qu'il engendre est grammatical dans tous les cas (on comprend aisément par exemple que des constituants de type SO ne peuvent être admis au sein des langues SVO que dans des cas bien particuliers, or le mécanisme proposé ci-dessus permet théoriquement de combiner n'importe quel couple de NP juxtaposés). Les règles croisées présentent donc un danger non-négligeable vis-à-vis des problèmes de surgénération.

### 3.2 Composition croisée : surgénérations

Reprenons l'exemple *I give a policeman a flower*. Il est clair que les étapes clés du calcul 3.1 pour la construction relative (en particulier, l'invocation de la composition croisée entre *give* et *a flower*) sont aussi bien applicables dans le cas standard, sans relative, rendant possible la dérivation de *\*I give a flower a policeman* alors que cet ordre est incorrect pour signifier « Je donne une fleur au policier. »

$$\frac{\frac{I}{NP} \quad \frac{\text{give}}{S \setminus NP / NP_{acc} / NP_{dat}} \quad \frac{\frac{\text{a flower}}{NP_{acc}}}{(S \setminus NP) \setminus (S \setminus NP / NP_{acc})} < T}{S \setminus NP / NP_{dat}} < B_{\times} \quad \frac{\text{a policeman}}{NP}$$

On voit que, hors des constructions relatives, la composition croisée induit une violation de la place fixe que doit occuper le datif en anglais.

On peut trouver de nombreux exemples de situations semblables, où la composition croisée entraîne des variantes prohibées de l'ordre des mots. Citons deux phrases données dans [2] :

- 1) Mauvais placement de modifieurs, *\*a powerful by Rivaldo shot* comme variante interdite de *a powerful shot by Rivaldo* :

$$\frac{\frac{\text{powerful}}{N/N} \quad \frac{\text{by Rivaldo}}{N \setminus N}}{N/N} < \mathbf{B}_\times \quad \frac{\text{shot}}{N} >$$

$$N!$$

2) Extraction de sujets emboîtés (en général interdite en anglais), comme dans la phrase incorrecte : *\*team that John knew that would defeat China* (pour dire : « l'équipe dont John savait qu'elle battrait la Chine ») (le « site d'extraction » -position normale avant extraction- du sujet est ici emboîté dans le domaine de *John knew...*

$$\frac{\frac{\text{team}}{N} \quad \frac{\frac{\text{that}}{N \setminus N / S \setminus NP} \quad \frac{\frac{\text{John knew that}}{S/S} \quad \frac{\text{would defeat China}}{S \setminus NP}}{S \setminus NP}}{N \setminus N}}{N!} <$$

$$> \mathbf{B}_\times$$

On voit que la règle  $>\mathbf{B}_\times$  rend possible ce genre d'extractions alors qu'elles sont prohibées en anglais.

[27] propose de résoudre ce type de problèmes en restreignant le domaine d'applicabilité des règles croisées, de façon à bloquer les cas indésirables. L'exemple 1) ci-dessus, par exemple, est traité en obligeant la règle de composition croisée à ne s'appliquer qu'à des catégories de type « prédicat », c'est à dire de type  $S\mathbb{S}$  (excluant donc  $N/N$  et  $N \setminus N$ , supprimant ainsi cette surgénération). L'exemple 2) sera traité en bannissant purement et simplement la règle  $>\mathbf{B}_\times$  de l'ensemble des règles de l'anglais.

Le problème du datif fixe (le policier et la fleur), lui, sera géré en introduisant un trait  $+/-\text{SHIFT}$  autorisant (pour le trait  $+\text{SHIFT}$ , interdisant pour le trait  $-\text{SHIFT}$ ) une catégorie à « permuer » via les règles compositions croisées, le «  $Z$  » des règles croisées (cf. ci-dessous) requérant un trait  $+\text{SHIFT}$ , et le datif des verbes ditransitif se voyant marquer d'un  $-\text{SHIFT}$  ( $S \setminus NP / NP / NP_{-\text{SHIFT}}$ ), bloquant la surgénération signalée. Mais ce faisant, l'analyse des constructions relatives 3.1 est perdue ! Il est alors nécessaire d'introduire une *deuxième* règle de composition croisée arrière spécifique au cas des relatives :

$$< \mathbf{B}_\times\text{-II} : Y/Z_{-\text{SHIFT}, +\text{ANT}} X \setminus Y \rightarrow X/Z_{-\text{SHIFT}, +\text{ANT}}$$

Où le trait  $\text{ANT}$  marque les arguments extraits. Il intervient dans le lexique comme suit : tous les noms du lexique sont marqués  $-\text{ANT}$ , les arguments des verbes sont sous-spécifiés pour ce trait (ex :  $S \setminus NP_{\pm \text{ANT}} / NP_{\pm \text{ANT}}$ ). L'argument «  $Z$  » de la  $<\mathbf{B}_\times\text{-II}$  portant le trait  $+\text{ANT}$  qui permettra de rétablir les analyses comme 3.1, tout en bloquant les surgénérations comme 3.2 (l'application sera bloquée car *a policeman* ne portera pas le trait  $+\text{ANT}$ ).

En résumé, après l'introduction des restrictions proposées par Steedman, l'ensemble des règles croisées devient :

- pas de règle  $>\mathbf{B}_\times$
- $<\mathbf{B}_\times\text{-I} : Y/Z_{+\text{SHIFT}} X \setminus Y \rightarrow X/Z_{+\text{SHIFT}}$
- $<\mathbf{B}_\times\text{-II} : Y/Z_{-\text{SHIFT}, +\text{ANT}} X \setminus Y \rightarrow X/Z_{-\text{SHIFT}, +\text{ANT}}$  avec  $X, Y$  obligatoirement de la forme  $S\mathbb{S}$

On aura pu déjà constater à travers ces quelques exemples que le maniement des restrictions peut s'avérer délicat. Souvent, en effet, les restrictions introduites s'avèrent trop fortes et engendrent en retour une sous-génération, et il est nécessaire soit d'affaiblir certaines restrictions du système pour écarter les sous-générations, soit d'introduire une nouvelle instanciation de ces règles (telle la règle  $<\mathbf{B}_\times\text{-II}$ ) pour couvrir certains cas particuliers... Il s'ensuit une duplication des règles néfaste à la concision du système. En fait, tout comme il était difficile de contrôler l'impact du degré de liberté introduit par les règles croisées, il n'est pas facile de calculer celui découlant de restrictions imposées *a posteriori* sur ce degré de liberté. Nous reviendrons en détail sur les limites des restrictions de règles dans l'introduction de la partie 4.

### 3.3 Étude du cas du français

Pour se convaincre du fait que ce type de phénomène n'est pas circonscrit à la seule langue anglaise, et pour évaluer l'étendue des mécanismes de « restrictions » (quelle que soit la forme prise par ceux-ci) qui peuvent

être nécessaires, nous avons construit un corpus de surgénérations engendrées par les règles croisées en français. Pour alléger les expressions, nous utiliserons parfois l'abréviation  $SV =_{def} S \setminus N$ .

n°	Type de surgénération	Phrase exemple	Cause
1	<b>Place du verbe principal dans les relatives</b>	<i>*Le chat qui déguste ronronne son lait</i>	$\frac{\frac{\text{Le chat qui déguste}}{NP/NP} \quad \frac{\text{ronronne}}{S \setminus NP}}{S/NP} < \mathbf{B}_x$
2	<b>Position relative des adjectifs</b>	<i>*Une belle rouge pomme</i>	$\frac{\frac{\text{belle}}{N/N} \quad \frac{\text{rouge}}{N \setminus N}}{N/N} < \mathbf{B}_x$
3	<b>Position relative des adjectifs</b> <sup>12</sup>	<i>**Une histoire sacrée éternelle</i>	$\frac{\frac{\text{sacrée}}{N/N} \quad \frac{\text{éternelle}}{N \setminus N}}{N/N} < \mathbf{B}_x$
4	<b>Adjectifs et relatives</b>	<i>*La jolie que tu vois femme</i>	$\frac{\frac{\text{jolie}}{N/N} \quad \frac{\text{que tu vois}}{N \setminus N}}{N/N} < \mathbf{B}_x$
5	<b>Négation : place des particules <i>ne</i> et <i>pas</i></b>	<i>*Je ne pas dors</i>	$\frac{\frac{\text{ne}}{SV/SV} \quad \frac{\text{pas}}{SV \setminus SV}}{SV/SV} < \mathbf{B}_x \quad \frac{\text{dors}}{SV} >$
6	<b>Négation : place des particules <i>ne</i> et <i>pas</i></b>	<i>*J'aime ne pas les épinards</i>	$\frac{\frac{\text{aime}}{SV/NP} \quad \frac{\frac{\text{ne}}{SV/SV} \quad \frac{\text{pas}}{SV \setminus SV}}{SV \setminus SV}}{SV/NP} > \mathbf{B}_x$
7	<b>Négation : <i>ne</i> et adverbess négatifs</b>	<i>*Je ne jamais dors</i>	$\frac{\frac{\text{ne}}{SV/SV} \quad \frac{\text{jamais}}{SV \setminus SV}}{SV/SV} < \mathbf{B}_x \quad \frac{\text{dors}}{SV} >$
8	<b>Place de l'adverbe <i>y</i></b>	<i>*Je vais y souvent</i>	$\frac{\frac{\text{vais}}{SV} \quad \frac{\frac{\text{y}}{SV/SV} \quad \frac{\text{souvent}}{SV \setminus SV}}{SV \setminus SV}}{SV} > \mathbf{B}_x$
9	<b>Place de l'adverbe <i>y</i></b>	<i>*Je y souvent vais</i>	$\frac{\frac{\text{y}}{SV/SV} \quad \frac{\text{souvent}}{SV \setminus SV}}{SV/SV} < \mathbf{B}_x \quad \frac{\text{vais}}{SV} >$
9bis	<b>Place de l'adverbe <i>y</i></b>	<i>*Je y souvent vais avec le ty-page alternatif vais : SV/PP, y : SV/(SV/PP)</i>	$\frac{\frac{\text{y}}{SV/(SV/PP)} \quad \frac{\text{souvent}}{SV \setminus SV}}{SV/(SV/PP)} < \mathbf{B}_x \quad \frac{\text{vais}}{SV/PP} >$
10	<b>Place des adverbess modifieurs de phrase</b>	<i>*Il hier dormait</i> <sup>13</sup>	$\frac{\frac{\text{hier}}{S/S} \quad \frac{\text{dormait}}{S \setminus NP}}{S \setminus NP} < \mathbf{B}_x$
11	<b>Place des adverbess modifieurs de phrase</b>	<i>*Jean ne hier dormait pas</i>	(comme ci-dessus)
12	<b>Place des adverbess modifieurs de phrase</b>	<i>*Jean ne dormait hier pas</i>	$\frac{\frac{\text{hier}}{S/S} \quad \frac{\text{pas}}{(S \setminus NP) \setminus (S \setminus NP)}}{(S \setminus NP) \setminus (S \setminus NP)} > \mathbf{B}_x^2$
13	<b>Préposition <i>de</i></b>	<i>*Un hangar tôle de ondulée (pour Un hangar de tôle ondulée)</i>	$\frac{\frac{\text{de}}{N \setminus N/N} \quad \frac{\text{ondulée}}{N \setminus N}}{N \setminus N/N} > \mathbf{B}_x$

14	<b>Verbes pronominaux</b>	<i>*La tendance se vite accélère</i>	$\frac{\frac{\text{se}}{\text{SV}/(\text{SV}/\text{NP})} \quad \frac{\text{vite}}{\text{SV}\backslash\text{SV}}}{\frac{\text{accélère}}{\text{SV}/(\text{SV}/\text{NP})}} < \mathbf{B}_\times \frac{\text{accélère}}{\text{SV}/\text{NP}} >$
15	<b>Verbes semi-auxiliaires et négation</b>	<i>**Je peux ne pas dormir. (permet l'interprétation incorrecte « Je ne peux pas dormir »)</i>	$\frac{\text{peux}}{\text{SV}/\text{SV}_{inf}} \quad \frac{\text{ne}}{\text{SV}/\text{SV}} \quad \frac{\text{pas}}{\text{SV}\backslash\text{SV}} > \mathbf{B}_\times$ $\frac{\text{ne pas}}{\text{SV}/\text{SV}_{inf}} < \mathbf{B}_\times$
16	<b>Qui constructeur de nom</b>	<i>**Qui dort dîne. (permet l'interprétation incorrecte « Qui dîne dort »)</i>	$\frac{\text{Qui}}{\text{NP}/\text{SV}} \quad \frac{\text{dort}}{\text{S}\backslash\text{NP}} > \mathbf{B}_\times \quad \frac{\text{dîne}}{\text{SV}} >$ $\frac{\text{dort dîne}}{\text{S}/\text{SV}} > \mathbf{B}_\times \quad \frac{\text{dîne}}{\text{SV}} >$
17	<b>Qui constructeur de nom</b>	<i>*Qui discutera mourra les ordres. (pour « Qui discutera les ordres mourra »)</i>	$\frac{\text{Qui}}{\text{NP}/\text{SV}} \quad \frac{\text{discutera}}{\text{SV}/\text{NP}} > \mathbf{B} \quad \frac{\text{mourra}}{\text{S}\backslash\text{NP}} < \mathbf{B}_\times$ $\frac{\text{discutera mourra}}{\text{NP}/\text{NP}} > \mathbf{B} \quad \frac{\text{mourra}}{\text{S}\backslash\text{NP}} < \mathbf{B}_\times$ $\frac{\text{discutera mourra}}{\text{S}/\text{NP}} < \mathbf{B}_\times$

TABLE 3: Surgénérations engendrées par les règles de compositions croisées  $> \mathbf{B}_\times$ ,  $< \mathbf{B}_\times$  en français.

Étudions rapidement les possibilités qu'une stratégie de restrictions de règles comme celles vues ci-dessus offrirait :

- Bannir la composition croisée avant des règles du français permettrait de bloquer les surgénérations 6,8,12,13,15 et 16 (mais, évidemment, nous priverait entre autres de l'analyse de la coordination 1 évoquée plus haut).
- Restreindre la composition croisée arrière aux seuls arguments de la forme S\$ bloquerait les surgénérations 1,2,3,4,17
- Les cas 5,7,9,10,11,14 posent plus de difficultés... Techniquement parlant, il « suffirait » d'interdire aux mots *ne*, *y*<sup>14</sup>, aux adverbes modificateurs de phrases (comme « hier »), d'accéder à la règle de composition croisée arrière... On pourrait par exemple créer une restriction *ad hoc* interdisant le premier terme de la composition croisée d'être un modifieur (de la forme X/X) ou introduire des traits bloquant les dérivations indésirées (à noter que le trait SHIFT de [27] ne peut être repris directement, les catégories « Z » correspondant aux catégories de *ne, se...* n'étant pas atomiques).

### 3.4 Autres cas de surgénération

Notons que toutes les surgénérations ne découlent pas des règles croisées et d'un phénomène de permutativité. Quoiqu'ils soient en général moins répandus, [2] relève quelques exemples de surgénérations mettant en jeu la seule règle de composition harmonique (et que l'auteur traite en ayant recours à sa modalité  $\star$  de « restriction de l'associativité », voir partie suivante). Nous retranscrivons ici l'un d'eux *\*The player that shoots and he misses* :

$$\frac{\text{The player that}}{\text{NP}/(\text{S}\backslash\text{NP})} \quad \frac{\text{shoots}}{\text{S}\backslash\text{NP}} \quad \frac{\text{and}}{\text{X}\backslash\text{X}/\text{X}} \quad \frac{\text{he misses}}{\text{S}} >$$

$$\frac{\text{shoots and he misses}}{\text{S}\backslash\text{NP}} < \mathbf{B}$$

$$\frac{\text{shoots and he misses}}{\text{NP}!} >$$

13. Avec sacré : N/N ; qui a un sens différent de sacré : N\N (qui serait attendu dans ce contexte)... L'utilisation du premier type (que rien ne permet à l'analyseur d'écarter *à priori*), peut donc engendrer l'interprétation incorrecte « une sacrée histoire... » (en plus de la correcte, « une histoire sacrée... »). Nous noterons \*\* ce type de situation.

13. Noter cependant que des phrases avec un sujet topicalisé, comme *Jean, hier, ne dormait pas* doivent être acceptées (on prêtera cependant attention à l'exemple 11) )

14. Sachant qu'il est impératif de maintenir l'accès des mots comme *pas, jamais, ceux-ci* ayant besoin de la composition croisée pour se combiner avec le verbe dans des phrases comme *Je n'aime pas les épinards*



On voit que la surgénération provient de la règle harmonique arrière  $\langle \mathbf{B}$  (notons cependant que cet exemple est fortement lié au mécanisme de traitement de la coordination utilisé ici, qui est peut-être à revoir dans son ensemble, [9]). Nous n'étudierons pas plus en détail ce type de surgénération dans ce rapport, cependant elles soulèvent des questions très importantes. Il conviendrait en effet de mener à bien une étude détaillée de ces phénomènes pouvant nécessiter de telles « restrictions de l'associativité », car la propriété d'associativité, d'un autre côté, est cruciale notamment pour pouvoir mener à bien une analyse incrémentale.

D'autres surgénérations découlent simplement d'insuffisances du système de base pour formuler simplement certaines contraintes, par exemple sur l'ordre des mots. Les modificateurs, notamment, souffrent de telles insuffisances. Prenons les adverbes *ne* et *y*, le type que ces adverbes se voient généralement attribuer en première approximation (SV/SV) les autorise à permuter librement par rapport à l'autre. Or, dans la réalité leur ordre est contraint, *ne* devant obligatoirement précéder *y* (comme on l'a vu dans l'analyse des clitiques de [15], cf. partie 2.3). Nous verrons dans la partie 6 des mécanismes permettant de gérer ce genre de restrictions.

## 4 La Grammaire Catégorielle Combinatoire Multi-Modale

Dans la partie précédente, nous avons vu comment les règles de composition croisées, de par l'altération de l'ordre canonique des mots qu'elles introduisent, peuvent déboucher sur un certain nombre de problèmes de surgénérations. Nous avons également vu comment il est possible de combattre ces surgénérations via l'utilisation de **restrictions de règles**. Bien qu'elle offre un moyen rapide de bloquer certaines constructions indésirées, cette stratégie n'est pas sans poser un certain nombre de difficultés, et ce pour plusieurs raisons. En premier lieu, les restrictions introduites pour traiter un phénomène particulier s'avèrent souvent trop fortes après-coup, car elles entraînent le blocage de constructions valides qui autrefois étaient reconnues par le modèle (ou bien, en tentant d'étendre la couverture du modèle, on se heurte à une restriction introduite auparavant), c'est-à-dire des sous-génération. Il est alors nécessaire de revenir sur les restrictions préalablement ajoutées en traitant ces cas de sous-génération tout en continuant de traiter les cas de surgénérations qui avaient motivé leur introduction, et c'est une tâche éminemment complexe, car on ne dispose d'aucun outil formel permettant d'évaluer l'impact des modifications apportées. Le défaut principal de cette approche est d'apporter une réponse globale (restriction d'une règle, voire son bannissement) à des problèmes locaux (ex. ordonnancement fixe d'une certaine catégorie de mots), pouvant mener à des généralisations abusives, et rendant l'expression des règles et de leurs restrictions de plus en plus alambiquées à mesure que l'on y intègre les différentes exceptions, exceptions aux exceptions...

Un autre reproche que l'on peut formuler est que cette stratégie abandonne l'idée de considérer l'ensemble des règles combinatoires comme un socle commun partagé par toutes les langues, l'expression des différentes restrictions rendant *de facto* cet ensemble différent pour chaque langue ([27] utilise par exemple, dans son analyse du Néerlandais, deux versions de composition harmonique avant  $\mathbf{>B}$ , chacune différente de la règle utilisée pour l'anglais ou le français).

On notera enfin la tendance de l'introduction des restrictions à multiplier le nombre de règles, témoins les règles  $\langle \mathbf{B}_x\text{-I}$  et  $\langle \mathbf{B}_x\text{-II}$  utilisées par [27].

Dans cette partie, nous allons étudier une solution alternative aux restrictions de règles, proposée par [2], consistant à contrôler l'accès aux règles combinatoires directement dans le lexique, via l'introduction d'un système de modalités s'appuyant sur le formalisme CTL, permettant aux catégories syntaxiques d'exprimer l'information auparavant portée par les restrictions de règles, ce qui, comme on va le voir permet de dépasser certains des obstacles mis en évidence ci-dessus.

### 4.1 Règles GCC et preuves CTL

Nous avons souligné dans la partie 2 le lien existant entre la notion de règle structurelle des CTL et les règles combinatoires de GCC. La différence entre les deux points de vue réside dans le fait que les CTL reposent sur deux mécanismes. D'une part, les règles logiques (introduction, élimination...) qui permettent (via différents *modes de compositions* unaires et binaire) de créer de nouvelles structures à partir de prémisses (par exemple en assemblant deux prémisses en une structure plus complexe), et les **règles structurelles** d'autre part, qui permettent de réorganiser les structures produites (e.g. réordonner le parenthésage, l'ordre des constituants).

Comme on l'a vu, les règles de réorganisation structurelle peuvent éventuellement être sensibles aux modalités qui marquent les liens de leur structure prémisses, et ainsi réaliser un contrôle précis de l'utilisation des *ressources linguistiques* (c.f. les concepts de la logique linéaire [13]).

En GCC, en revanche, il n'y a pas de mécanisme de réorganisation, car les règles GCC n'ont que le pouvoir de *construire* les structures. Noter également que les règles GCC ne peuvent s'appliquer qu'à des unités linguistiques effectivement présentes, contrairement à CTL qui s'autorise des mécanismes d'introduction et élimination d'hypothèses. par ailleurs, GCC s'autorise à stipuler une plus grande variété de règles de construction (levée de type, composition harmonique, composition croisée) qui en réalité, de part les effets qu'elles induisent, sous-entendent *implicitement* des réorganisations structurelles (mais celles-ci ne sont pas isolables, et sont « compilées » dans le processus insécable de l'application de la règle). Étudions par exemple les différents mécanismes mis en jeu dans chacun des deux systèmes pour réaliser la composition entre semi-auxiliaire et un verbe transitif (e.g. « aimer » et « lire » dans la phrase *Felipe aime lire des bandes dessinées*) :

- Analyse du constituant *aime lire* : GCC

$$\frac{\frac{\text{aime}}{\text{SV/SV}_i} \quad \frac{\text{lire}}{\text{SV}_i/\text{NP}}}{\text{SV/NP}} >\mathbf{B}$$

- Analyse du constituant *aime lire* : CTL

$$\frac{\frac{\text{aime} \vdash \text{SV/SV}_i \quad \frac{\frac{\text{lire} \vdash \text{SV}_i/\text{NP} \quad x \vdash \text{NP}}{\text{lire} \circ x \vdash \text{SV}_i} \text{E/}}{\text{aime} \circ (\text{lire} \circ x) \vdash \text{SV}} \text{E/}}{\frac{(\text{aime} \circ \text{lire}) \circ x \vdash \text{SV}}{\text{aime} \circ \text{lire} \vdash \text{SV/NP}} \text{I/}} \mathbf{RA}$$

On constate que dans le cas de GCC, cette combinaison correspond à une simple composition harmonique, là où l'analyse CTL requiert plusieurs étapes et mécanismes (introduction d'hypothèse, règle structurelle d'associativité, éliminations et introduction de slash). La preuve CTL peut donc être vue ici comme une sorte de « microscope » (expression empruntée à [14]) permettant de révéler et séparer les différents ingrédients condensés dans la règle  $>\mathbf{B}$  de GCC (notamment, le recours à la règle d'associativité).

En résumé, les preuves CTL nécessitent davantage d'étapes, permettant un découpage fin des différents éléments qui sous-tendent une analyse (introduction d'hypothèses, réorganisation structurelle...), éléments qui, dans le cas des règles GCC, sont « cachés » car compilés dans une seule règle combinatoire, et effectués en une seule étape. Inversement, les règles GCC constituent un moyen de *compiler* les preuves CTL en des analyses GCC plus concises et faisant appel à des mécanismes moins puissants (notamment, sans introduction explicite d'hypothèses), donc possédant des propriétés algorithmiques plus attractives.

Cette connexion fondamentale entre GCC et CTL est à la base de la Grammaire Combinatoire Multi-Modale de [2], qui introduit un contrôle lexicalisé des phénomènes de surgénération. Si les preuves CTL permettent de révéler les différentes composantes « compilées » dans les règles GCC (notamment, les règles structurelles), elles nous rendent donc également capables d'isoler celle(s) qui sont à l'origine des surgénérationes que certaines des règles GCC engendrent. L'idée est donc de simuler les règles GCC via CTL, d'isoler les causes des surgénérationes puis d'exploiter le système de modalité de CTL pour contrôler l'accès à ces causes, et enfin, de « compiler » les preuves CTL obtenues en règles combinatoires GCC *modales*. La section suivante présente la mise en pratique de cette idée et aboutit au système de modalités de [2].

## 4.2 Analyse de Baldrige : introduction des modalités

Comme les surgénérationes que nous avons étudiées concernaient essentiellement les règles  $>\mathbf{B}_\times$ ,  $<\mathbf{B}_\times$ , nous présenterons ici en priorité les développements théoriques concernant ces règles et le contrôle de la permutativité (les démonstrations associées aux autres règles utilisant les mêmes idées).

En vue de simuler les règles GCC à l'aide des règles logiques et structurelles de CTL, [2] commence par remarquer que les règles GCC sont sensibles à la directionnalité des slash des catégories qui la composent. C'est en partie évident, car les opérateurs consomment leur argument dans la direction définie par leur slash, mais ce n'est pas tout. Observons la règle croisée arrière par exemple :

$$\langle \mathbf{B}_\times : Y/Z \quad X \setminus Y \rightarrow X/Z$$

Le slash du  $Z$  ne consomme pas son argument au cours de la combinaison, il est inerte, cependant la règle croisée ne peut s'appliquer qu'à la condition que sa direction soit inverse à celle du slash actif,  $\setminus Y$ . Pour traduire ceci en CTL, il faut rendre les règles structurelles sensibles à la directionnalité des slashes, et pour ce faire il est nécessaire (selon [2], cependant nous discuterons cette « nécessité » en fin de partie) d'introduire deux modalités servant à garder trace de la direction du slash GCC ayant, après transposition en CTL, servi à former un produit  $A \circ B$ . Ces deux modes de composition sont notés :  $M_1 = \{\triangleright, \triangleleft\}$

Nous sommes maintenant en mesure de développer la preuve CTL correspondant à la règle de composition croisée (ici, arrière) :

$$\frac{\frac{\frac{\Gamma \vdash Y/\triangleright Z \quad x \vdash Z}{(\Gamma \circ_{\triangleright} x) \vdash Y} \text{ /}_\triangleright \mathbf{E} \quad \Delta \vdash X \setminus \triangleleft Y}{((\Gamma \circ_{\triangleright} x) \circ_{\triangleleft} \Delta) \vdash X} \text{ /}_\triangleleft \mathbf{E}}{\frac{((\Gamma \circ_{\triangleleft} \Delta) \circ_{\triangleright} x) \vdash X}{(\Gamma \circ_{\triangleleft} \Delta) \vdash X/\triangleright Z} \text{ /}_\triangleright \mathbf{I}}{\text{MRP}}$$

Crucialement, on remarque que cette preuve utilise une règle structurelle « Mixed Right Permutation » (MRP), que nous recopions ci-dessous. C'est précisément en cette règle de réorganisation qu'est contenue ce que nous avons auparavant appelé le « degré de liberté » introduit par la règle de composition croisée, la *permutation* par rapport à l'ordre canonique des mots, et c'est donc sur cette règle qu'il faut agir pour pouvoir contrôler les phénomènes de permutation. [2] introduit donc une modalité  $\times$  qui contrôle l'accès à cette règle structurelle (elle se combine aux deux modalités précédentes, créant deux nouvelles modalités  $M_2 = \{\triangleleft \times, \times \triangleright\}$ ), laquelle devient :

$$\frac{((\Delta_a \circ_{\times \triangleright} \Delta_b) \circ_{\triangleleft \times} \Delta_c) \vdash X}{((\Delta_a \circ_{\triangleleft \times} \Delta_c) \circ_{\times \triangleright} \Delta_b) \vdash X} \text{MRP}_\times$$

Seules les structures formées avec le mode de composition  $\times$  (avec les directionnalités adéquates) pourront ainsi accéder à la règle structurelle. On peut voir que par conséquent, tous les slashes et produits de la preuve 4.3 devront porter ladite modalité pour que la preuve tienne toujours. Ceci étant établi, nous pouvons maintenant condenser la preuve 4.3 en une règle GCC dont les slashes porteront les modes adéquats ; il suffit pour cela de ne retenir que les prémisses et la conclusion :

$$\langle \mathbf{B}_{\times \text{GCC-MM}} : Y/\times \triangleright Z \quad X \setminus \triangleleft \times Y \rightarrow X/\times \triangleright Z$$

On peut ainsi à loisir autoriser ou interdire l'accès d'une catégorie particulière à la règle de composition croisée arrière, il suffira pour ce faire de donner ou retirer à son slash la modalité  $\times$  (voir exemples ci-après).

Une analyse similaire nous aurait montré que la règle  $\triangleright \mathbf{B}_\times$  repose de même sur une règle structurelle très semblable à la précédente « Mixed Left Permutation » (MLP) :

$$\frac{(\Delta_a \circ_{\times \triangleright} (\Delta_b \circ_{\triangleleft \times} \Delta_c)) \vdash X}{(\Delta_b \circ_{\triangleleft \times} (\Delta_a \circ_{\times \triangleright} \Delta_c)) \vdash X} \text{MLP}_\times$$

Règle que l'on peut comme auparavant contrôler via une modalité. On peut choisir, comme dans [2] d'utiliser la même modalité que précédemment,  $\times$ , (nous avons directement écrit la version modalisée de la règle structurelle), menant *in fine* à la règle GCC-MM suivante :

$$\langle \mathbf{B}_{\times \text{GCC-MM}} : X/\times \triangleright Y \quad Y \setminus \triangleleft \times Z \rightarrow X/\times \triangleright Z$$

Le même travail est effectué pour les règles structurelles d'associativité sous-tendant les règles de composition harmonique, on choisit<sup>15</sup> de créer une modalité  $\diamond$  (non divisée en deux versions directionnelles) contrôlant cette règle, menant aux règles GCC

15. Bien noter que la plupart du temps, plusieurs choix de modalités sont possibles pour contrôler les règles structurelles en jeu, le système de modalité proposé par [2] n'est donc qu'une possibilité parmi beaucoup d'autres.

$$\begin{aligned} >\mathbf{B}_{GCC-MM} : X/\circ Y \quad Y/\circ Z \rightarrow X/\circ Z \\ <\mathbf{B}_{GCC-MM} : Y\backslash\circ Z \quad X\backslash\circ Y \rightarrow X\backslash\circ Z \end{aligned}$$

[2] utilise encore deux modalités :  $\star$  (la modalité « de base », qui n'a accès qu'à la règle d'application), et  $\cdot$  (modalité qui hérite des propriétés de toutes les autres). Au total, la GCC-MM introduit 6 modalités :

$$M_{MM-GCC} = \{\star, \diamond, \llcorner, \lrcorner, \triangleleft, \triangleright, \cdot\}$$

Ces modalités ne sont pas indépendantes, et sont structurées selon l'arbre d'héritage suivant :

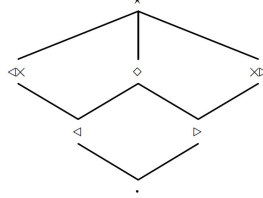


FIGURE 1 – Les relations d'héritage entre les modalités définies par Baldridge dans [2].

Mentionnons en passant une convention d'écriture employée par [2] : dans le cas où l'information de directionnalité contenue dans les modalités  $\{\llcorner, \lrcorner, \triangleleft, \triangleright\}$  coïncide avec le slash qui la porte, celle-ci n'est pas marquée, ainsi  $\lrcorner, \lrcorner, \triangleleft, \triangleleft$  sont respectivement abrégés en  $/, /, \backslash, \backslash$ .

Le lecteur pourra se demander comment des slashes comme  $\triangleleft$ , dont la direction du slash et de la modalité sont opposées, peuvent survenir dans le lexique. Ils sont en fait une conséquence de la règle de levée de type, comme on le voit dans la preuve CTL associée ci-dessous (on note  $\circ_i$  une modalité quelconque) :

$$\frac{\frac{\Delta \vdash X \quad x \vdash Y \backslash_i X}{\Delta \circ_i x \vdash Y} \backslash_i E}{\Delta \vdash Y /_i (Y \backslash_i X)} /_i I$$

(On notera que la démonstration de la levée de type, ne fait appel à aucune règle structurelle, mais repose uniquement sur les règles logiques, elle est une conséquence directe de la règle d'introduction de slash.) On voit ainsi que les deux slashes créés doivent impérativement porter la même modalité (laquelle peut être quelconque). Les règles de levée de type (avant et arrière, la démonstration associée à la règle arrière étant similaire) GCC-MM seront ainsi formulées de la manière suivante :

$$\begin{aligned} X &\rightarrow Y /_i (X \backslash_i Y) \\ X &\rightarrow Y \backslash_i (X /_i Y) \end{aligned}$$

Où la variable  $i$  exprime le fait que les deux slashes doivent porter la même modalité. C'est par cette règle que peuvent se créer des catégories comportant des slashes de direction opposée à celle de leur modalité.

Voyons maintenant comment ces modalités peuvent être utilisées pour se passer de la notion de restriction de règle, en déléguant au lexique la capacité de contrôle auparavant dévolue à ces restrictions.

### 4.3 Applications linguistiques : gain par rapport aux analyses précédentes, limites

Maintenant que l'accès aux règles combinatoires est contrôlé par les modalités, il est possible de spécifier des catégories ne pouvant accéder qu'à une partie de l'ensemble des règles, et nous soulager du besoin d'accumuler des restrictions spécifiées au niveau des règles elles-mêmes. Reprenons les exemples de surgénérations issues des règles croisées, par exemple *a powerful by Rivaldo shot*. Nous avons mis en évidence le fait que cette surgénération découlait de la composition croisée entre *powerful* et *by Rivaldo*. En attribuant à *powerful* la catégorie  $N/\circ N$ , qui n'hérite pas de la modalité  $\times$  autorisant l'accès aux règles croisées, on interdit aux adjectifs anglais de recourir aux règles croisées, exprimant le fait qu'aucun modifieur arrière (ex : *by Rivaldo*) ne peut s'intercaler entre eux et le nom qu'ils qualifient (cas de  $<\mathbf{B}_\times$ ), et qu'eux-mêmes ne peuvent s'intercaler entre leur nom et un modifieur arrière de ce nom (cas de  $>\mathbf{B}_\times$ ).

De même, la surgénération *the team that John knew that would defeat China* peut être supprimée en donnant au type de *that* (celui correspondant aux constructions comme « John knew that... ») la catégorie  $S/_\circ S$ .

On peut prendre des mesures similaires en ce qui concerne notre corpus d'exemples français. Les exemples 2,3,4,13 sur la position des adjectifs peuvent être traités de façon similaire au cas anglais, en attribuant aux adjectifs la catégorie  $N \setminus_\circ N$  ( $N/_\circ N$  pour les adjectifs avants,  $N \setminus_\circ N/N$  à la préposition *de* de l'exemple 13). Les exemples 5,6,7,8,9,10,11,12,14,15 cités précédemment peuvent maintenant être écartée de façon beaucoup plus souple (i.e. sans supposer de restriction de portée globale, et sans rendre alambiquée l'expression des règles de composition croisée), : il suffit de faire porter aux modifieurs avant de verbe comme *ne*, *y* la catégorie  $SV/_\circ SV$  (en première approximation... une analyse précise de ces mots, comme on l'a vu en 2.3, relevant en fait du traitement des clitiques - voir partie 6), aux « adverbes déterminants de phrase » la catégorie  $S/_\circ S$ , aux pronoms réfléchis  $SV/_\circ(SV/NP)$ .

L'exemple 16 peut être traité en restreignant le mode du type de *qui* en  $NP/_\circ SV$ .

Les exemples 1 et 17, cependant, soulèvent plus de questions. Reprenons la surgénération *Le chat qui déguste ronronne son lait* :

$$\frac{\frac{\text{Le chat qui}}{NP/(S \setminus NP)} \quad \frac{\text{déguste}}{S \setminus NP/NP}}{NP/NP} > \mathbf{B} \quad \frac{\text{ronronne}}{S \setminus NP} < \mathbf{B}_\times$$

$$\frac{\quad}{S/NP}$$

Le slash du type de *Le chat qui déguste* :  $NP/NP$  provient de l'argument du verbe transitif *ronronner* :  $SV/NP$  (la situation est similaire dans l'exemple 17). Ce slash doit absolument pouvoir accéder à la composition croisée pour que l'analyse de la place de l'adverbe 3.1 tienne toujours. Ne reste donc, dans le présent système, que la possibilité de restreindre le slash du sujet dans le verbe intransitif  $S \setminus_\circ NP$ .

Quoi que nous ne voyions pas (pour le moment<sup>16</sup>) de sous-génération évidente engendrée par l'introduction de cette modalité, on peut cependant relever un danger de généralisation abusive, car elle s'applique sur le slash d'une catégorie fondamentale, celle du verbe intransitif (et donc par extension sur tous les autres verbes), et son impact sur le modèle est plus difficilement cernable. Ce sentiment est renforcé par le fait que nous avons été contraints d'attribuer cette modalité par défaut d'un autre moyen de bloquer  $< \mathbf{B}_\times$ , sans réel argument linguistique autre que le besoin de bloquer la surgénération elle-même. On notera également que l'ancienne restriction de la règle  $< \mathbf{B}_\times$  (obligeant ses prémisses à être de la forme  $S\$$ ), quoique non moins contingente, ne comportait pas ce problème, car c'était la prémisse de gauche,  $NP/NP$ , qui bloquait la combinaison. Dans ce contexte, il apparaît clairement que la solution des modalités de slashes est moins expressive que les restrictions de règles, car elle ne permet pas d'accéder aux propriétés de la tête ( $NP/NP$ ) dans la catégorie précédente par exemple), et de bloquer ou autoriser la combinaison en conséquence. Voir la discussion 4.4 sur les avantages et limites de GCC-MM. L'intégration des modalités unaires que nous allons maintenant présenter peut cependant permettre de dépasser ces limitations.

Les exemples 3.1 et 3.2 nécessitent eux aussi des mécanismes plus élaborés que les seules modalités de slash. Il faut d'une part modéliser l'agrammaticalité (en anglais) des constructions Verbe-Objet Accusatif-Objet Datif (3.1), et la nécessité de devoir quand même pouvoir combiner un verbe ditransitif directement avec son accusatif dans le cas des constructions relatives (3.2). On se souvient que [27] introduit une deuxième règle de composition croisée (3.2), interagissant avec deux traits, ANT (qui traduit l'extraction de l'objet, donc le fait qu'il doit être consommé par un opérateur d'ordre supérieur, comme que :  $NP/(S/NP)$  ) SHIFT (qui autorise la permutation) pour sortir de l'interblocage entre ces deux exemples. Une telle duplication est également (*a priori*) inévitable avec GCC-MM : il importe que *give* et *flower* puissent se combiner par composition croisée (rejetant toute éventuelle tentation d'attribuer aux verbes ditransitifs la catégorie  $S/NP/_\circ NP$ ). Mais il importe que cette combinaison soit spéciale, et laisse une trace (ici, le trait +ANT) de façon à ce que l'argument « datif » de son produit soit obligatoirement extrait ( $S/NP_{+ANT}$ ) et ne puisse mener à la surgénération 3.2.

16. On pourrait par exemple imaginer d'utiliser cette règle croisée pour analyser des constructions avec incise, comme Jean, hier,<sub>S/S</sub> dormait  $S \setminus NP$

Pour continuer de disposer de l'interprétation CTL des preuves GCC-MM, il est nécessaire ici de faire intervenir les modalités unaires (voir la partie 2). On se souvient que les règles structurales peuvent être définies de façon à être sensibles aux modalités, binaires comme unaires, qui composent les structures mises en jeu, permettant d'obtenir l'effet recherché. L'équivalent d'une catégorie portant le trait  $+ANT$  sera une structure enfermée à l'intérieur du crochet structurel  $\langle \rangle_{ant}$ .

[2] remarque qu'il peut se passer du trait SHIFT en exploitant la directionnalité de ses modalités. Le principe est d'utiliser dans le lexique des slashes de direction contraire à celle de leur modalité (tels ceux issus de levées de type), et de s'en servir comme déclencheurs des règles  $\langle \mathbf{B}_\times^{ant}$  spéciales. Il définit ainsi trois règles  $\langle \mathbf{B}_\times^{ant}$  pour couvrir toutes les possibilités de directionnalité :

- a)  $\langle \mathbf{B}_\times^{ant} \text{ GCC-MM} : Y /_\times Z_{+ANT} \quad X \setminus_{\triangleright} Y \rightarrow X /_\times Z_{+ANT}$
- b)  $\langle \mathbf{B}_\times^{ant} \text{ GCC-MM} : Y /_{\triangleleft} Z_{+ANT} \quad X \setminus_\times Y \rightarrow X /_{\triangleleft} Z_{+ANT}$
- c)  $\langle \mathbf{B}_\times^{ant} \text{ GCC-MM} : Y \setminus_{\triangleleft} Z_{+ANT} \quad X \setminus_{\triangleright} Y \rightarrow X /_{\triangleleft} Z_{+ANT}$

Nous donnons ci-dessous la preuve CTL correspondant à la première règle (noter l'intervention de la règle structurale AGRP-I -Antecedent Governed Right Permutation- sensible à la modalité unaire  $\langle \rangle_{ant}$ ) :

$$\frac{\frac{\frac{\Gamma \vdash Y /_{\triangleleft} Z \quad \frac{x \vdash \square_{ant}^\downarrow Z}{\langle x \rangle_{ant} \vdash Z} \square_{ant}^\downarrow E}{(\Gamma \circ_{\triangleleft} \langle x \rangle_{ant}) \vdash Y} /_{\triangleright} E \quad \Delta \vdash X \setminus_\times Y}{((\Gamma \circ_{\triangleleft} \langle x \rangle_{ant}) \circ_{\triangleleft} \Delta) \vdash X} \setminus_\times E}{\frac{x_2 \vdash \diamond_{ant} \square_{ant}^\downarrow Z \quad ((\Gamma \circ_{\triangleleft} \Delta) \circ_{\triangleleft} \langle x \rangle_{ant}) \vdash X}{((\Gamma \circ_{\triangleleft} \Delta) \circ_{\triangleleft} x_2) \vdash X} \diamond_{ant} E} \text{ AGRP-I}}{\frac{((\Gamma \circ_{\triangleleft} \Delta) \circ_{\triangleleft} x_2) \vdash X}{(\Gamma \circ_{\triangleleft} \Delta) \vdash X /_{\triangleleft} \diamond_{ant} \square_{ant}^\downarrow Z} /_{\triangleleft} I} \diamond_{ant} E$$

Et attribue, par exemple, aux verbes transitifs comme *give* la catégorie :  $S/NP_{acc/\triangleleft} NP_{dat}$ , ce qui force l'utilisation des règles  $\langle \mathbf{B}_\times$  spéciales.

Cependant, on pourra pointer ce choix comme discutable. Compte tenu de l'argumentation de l'auteur, cette solution apparaît plus comme un moyen technique de « se passer du trait +SHIFT » en utilisant un « degré de liberté » du système (la directionnalité des modes), sans que l'interprétation linguistique, ni même le lien avec la notion de directionnalité, soient réellement clairs. On note que l'introduction même de ces modalités directionnelles mène à une duplication assez dérangeante des règles combinatoires, mais nécessaire pour énumérer les situations possibles.

En outre, ce « degré de liberté » n'est pas suffisant pour résoudre certaines situations de blocage entre sous- et surgénération. [2] donne les exemples suivants :

1. *The player that I [read a book about]<sub>SV/\triangleleft NP</sub> yesterday<sub>SV\setminus SV</sub>* (le joueur sur lequel j'ai lu un livre)

Ce type de construction (que l'anglais est une des rares langues à autoriser) impose, dans le présent formalisme, d'utiliser le slash  $/_{\triangleleft}$  de façon à permettre, dans cette situation d'extraction et seulement dans cette situation, la combinaison croisée arrière spéciale (introduisant un +ANT) avec *yesterday*. Cependant, ceci mène également à la surgénération représentée par l'exemple suivant.

2. *\*The player that [a book about]<sub>NP/\triangleleft NP</sub> [astonished me]<sub>SV\setminus NP</sub>* ( $\sim$  le joueur dont un livre sur lui m'a stupéfié)

Cette phrase, qui diffère de la précédente par le fait que c'est un sujet qui est extrait, est incorrecte et doit être bloquée. Compte tenu de l'exemple précédent, le seul degré de liberté restant est d'agir sur le slash de l'argument « sujet » de *astonished* :  $S\setminus NP/NP$ , donc en définitive sur celui des verbes intransitifs  $S\setminus NP$  (il est intéressant de remarquer que nous avons déjà envisagé cette restriction pour gérer la génération 1) de notre corpus d'exemple en français). Ici, la seule possibilité est de leur attribuer la catégorie  $S\setminus_{\triangleright} NP$ . Mais ceci bloquerait ensuite les extractions de sujets dans les constructions où elles sont permises, comme celle de la phrase suivante.

3. *The team that John [knew]<sub>S/S</sub> [would defeat China]<sub>S\setminus\_{\triangleright} NP</sub>* (l'équipe dont John savait qu'elle battrait la Chine)

La restriction introduite à l'étape précédente bloque cette phrase pourtant correcte, il faudrait ici que la composition croisée avant puisse s'appliquer pour poursuivre l'analyse, et nous avons épuisé les « degrés de liberté » du modèle.

D'autres degrés de liberté seront donc nécessaires, qui devront être introduits par un autre biais que celui des directionnalités. En cela cette solution ne semble pas optimale vis-à-vis du quatrième critère (robustesse aux extensions) que nous avons identifié dans l'introduction de ce rapport.

En définitive, c'est l'utilité même de l'intégration de l'information de directionnalité dans ces modalités qui peut être remise en cause. Leur introduction, comme on l'a vu, reposait sur la volonté de transcrire le fait que les règles GCC étaient sensibles aux directions des slash via des règles structurelles CTL également sensibles à ce paramètre. Cependant, ce raisonnement a eu des conséquences inattendues : la règle 4.2 ne couvre pas toutes les situations qui relevaient de  $\langle \mathbf{B}_\times \rangle$  dans GCC, car les catégories issues d'une levée de type (comme c'est le cas dans l'exemple 1) ne peuvent en être les prémisses, introduisant une contrainte inédite qui mène à introduire trois nouvelles règles pour traiter ces cas. L'intégration de l'information de directionnalité dans les modalités semble donc déboucher sur une certaine complexité (introduction de trois nouvelles règles au lieu d'une) assez peu justifiée au regard des services rendus (un « degré de liberté » supplémentaire).

C'est en fait la nature de la relation entre GCC-MM et CTL qui doit être précisée. S'il est intéressant d'analyser les règles combinatoires GCC-MM comme des théorèmes CTL afin de les analyser plus en profondeur, tenter à l'inverse d'intégrer certaines contraintes spécifiques à GCC dans CTL (telles les contraintes de directionnalité) est très susceptible de se solder d'un échec. En effet, il est fort probable que, compte tenu des mécanismes supplémentaires dont dispose CTL -introduction d'hypothèse, introduction de slash...-, les règles structurelles permettront de dériver d'autres théorèmes que les règles qu'elles sont censées « simuler » (cf. [2]), et la simulation dépassera immanquablement son objet. Ainsi, la pertinence de l'introduction de contraintes de GCC dans CTL semble limitée. Voir la partie suivante qui reprend cette discussion concernant relation entre CTL et GCC-MM.

De fait, nous avons noté d'ailleurs que dans les publications plus récentes de l'auteur ([3], [14], [28], [4]), la notion de directionnalité n'était plus intégrée (sans qu'il soit clairement établi, à notre connaissance, si cela constituait une simplification de [2] à but pédagogique ou une réelle révision du système) aboutissant à une simplification du système : l'ensemble des modalités se réduit à  $M_{MM-GCC} = \{\star, \diamond, \times, \cdot\}$  et l'arbre d'héritage à :

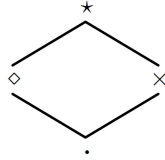


FIGURE 2 – Les relations d'héritage entre les modalités définies par Baldrige dans [3].

À notre sens, les pistes les plus prometteuses pour remplacer le « degré de liberté » de la directionnalité résident sans doute dans la définition de nouvelles règles, faisant éventuellement intervenir de nouvelles modalités, exprimant plus explicitement les cas d'exception par rapport au cas de base défini par les modalités. Par exemple :

$$X/_\times Y \ Y \backslash_\diamond Z_{+ANT} \rightarrow X \backslash_\diamond Z_{+ANT}$$

L'intérêt d'une telle formulation (semblable aux propositions de règles supplémentaires introduisibles qu'explore [2]) est la possibilité de l'interpréter comme la formulation explicite d'une exception à la restriction exprimée (ici) par la modalité  $\diamond$ , les conditions d'application de cette exception étant déterminées par l'intervention de modalités unaires telles ANT. Si des surgénérations sont à craindre, on peut envisager la création d'une modalité spéciale  $\times!$  dont il suffirait qu'une seule des prémisses la porte sur son slash pour que la composition croisée soit possible, permettant de modéliser facilement des informations linguistiques comme « en anglais, l'extraction de sujets de clauses emboîtées n'est *en général* pas possible » (catégorie  $S \backslash_\diamond NP$ ) « *sauf* lorsque cette clause est introduite par des verbes comme *know* » ( $S/_\times! S$ , forçant l'application de la règle croisée malgré la présence de  $\diamond$  sur l'autre slash). On pourrait même envisager (en se rappelant des stratégies utilisées dans 2.3 et 6) d'associer plusieurs « niveaux de force » aux modalités (une modalité d'un « niveau de force supérieure »

primerait sur l'autre pour déterminer l'applicabilité d'une règle binaire), ce pour gérer d'éventuelles situations complexes faisant apparaître des « exceptions aux exceptions »).

## 4.4 Bilan

Pour conclure quant aux apports et limites de GCC-MM, il convient d'identifier *deux niveaux* dans les idées que développe ce formalisme. Ce que nous appellerons le « premier niveau », plus abstrait, consiste en une *méthode* qui part du constat de la trop grande rigidité des restrictions de règles telles que celles de [27], et aboutit à leur remplacement par un système de modalités, importées du formalisme CTL (vu comme un « microscope » aidant à la conception des règles) et permettant d'exprimer cette notion de *restriction* directement dans le lexique. Ceci, comme on l'a vu, constitue un avantage non-négligeable : les modalités diffusent dans le lexique une complexité qui auparavant s'accumulait dans les règles combinatoires, permettant de rendre à celles-ci un caractère universel. L'ensemble de règles combinatoires est commun à toutes les langues, tandis que c'est le lexique qui instancie les catégories syntaxiques *et* les restrictions d'application des règles propres à une langue particulière. Ce système permet en outre de traiter plus facilement les exceptions, permettant d'agir de façon locale et « incrémentale » (i.e. sans devoir revenir sur une restriction définie préalablement par exemple) en introduisant des catégories portant une certaine modalité, et d'éviter certaines généralisations abusives auxquelles les restrictions de règles globales nous rendaient plus enclins.

L'universalité de l'ensemble des règles doit cependant être nuancée si l'on prévoit l'intégration de modalités CTL plus avancées. [15] utilise par exemple dans son analyse des clitiques du français, le mode de composition  $\circ_{ca}$  (pour « clitic-attachment »), qui exprime, comme on l'a vu, le fait que le lien entre un clitique et son verbe possède une « force » supérieure (justifiée par le fait que le clitique est en quelque sorte une forme intermédiaire entre le mot -morphème- indépendant et l'affixe), qui possède des propriétés particulières exprimées par des règles structurelles qui lui sont propres. Le statut des règles GCC-MM découlant de ces règles, et leur position par rapport à l'idéal d'universalité de l'ensemble de règles devra être évalué plus précisément.

À noter que l'interaction entre les différentes modalités peut conduire à une duplication des règles pour énumérer toutes les combinaisons possibles. Ainsi, il semble tout à fait indispensable d'organiser l'ensemble des modalités selon une structure d'héritage (telle celle proposée par [2]) pour éliminer les redondances entre les formulations des différentes règles. Cependant il faudra prêter une attention particulière à l'interprétation CTL de telles modalités structurées, ce système n'utilisant pas ([2]), dans ses formes les plus courantes, le concept d'unification, et ne fournit pas de moyen direct d'exprimer de telles relations d'héritage.

Enfin, si les avantages de l'utilisation des modalités sur les restrictions de règles semblent bien établis, on peut se demander cependant si toute analyse utilisant les restrictions de règles peut être effectivement transcrite en une analyse modale sans restrictions. [17], après avoir démontré que la puissance générative de GCC sans restrictions est strictement inférieure à celle de GCC avec restrictions de règles, étudie une certaine version de grammaire catégorielle multi-modale et démontre que celle-ci possède également une puissance générative strictement inférieure à celle de GCC avec restrictions de règles (cependant cette version est différente de celle de Baldrige, ne permettant donc pas de se prononcer quant à cette dernière). Il serait donc intéressant d'étudier les positions d'autres systèmes modaux (celui de [2] notamment) par rapport à GCC avec restrictions, en terme de puissance générative, et d'établir s'il est possible d'en construire un de puissance équivalente.

Le second niveau est le système particulier de modalités que propose [2]. On a vu que celui-ci permettait de modéliser un certain nombre de phénomènes linguistiques et de se passer effectivement de la notion de restriction, mais que certains obstacles demeurent et que les degrés de liberté du système ne sont pas suffisants pour gérer la complexité de certains phénomènes sans faire de concession à la surgénération (ou à la sous-génération). Voir par exemple les contraintes intriquées des trois phrases citées dans la section précédente. Nous avons en outre vu qu'il était sans doute souhaitable d'abandonner l'idée d'intégrer la notion de directionnalité aux modalités, les facilités apportées en terme de modélisation linguistique apparaissant assez négligeables au regard de la complexification induite du système.

Enfin, nous avons remarqué que le recours à la modalité  $\star$  introduisant une restriction de l'associativité n'est pas sans soulever certaines questions, car elle menace directement la possibilité d'utiliser le modèle pour réaliser une analyse incrémentale (par ailleurs identifiée comme désirable dans plusieurs articles utilisant GCC



ou GCC-MM [27], [19], [14]). Il peut être donc nécessaire de rechercher des solutions alternatives pour gérer les analyses utilisant ce recours.

En définitive, le système de modalités introduit par [2], de l’avis même de l’auteur, ne doit en aucun cas être considéré comme définitif. Les règles structurelles de CTL permettent de modéliser une grande variété de comportements, pouvant ensuite fonder une grande variété de systèmes modaux à la GCC-MM, mais plus élaborés que la première proposition de [2], et atteignant une meilleure couverture de la langue.

Dans les deux dernières parties, nous nous concentrerons sur le premier aspect, i.e. la méthode sous-jacente à la conception de GCC-MM. La partie suivante consiste en la poursuite de la discussion relative à la position « des » GCC-MM (sous-entendu, GCC-MM et ses éventuelles extensions basées sur la même méthode) par rapport à leur base CTL, et des pistes envisageables pour mettre en application et interpréter la méthode de transposition ébauchée par [2] dans un cadre plus large.

## 5 Fonder les extensions de GCC-MM par CTL

Dans cette partie, nous présentons les grandes lignes d’une démarche générale visant à se baser sur le formalisme CTL afin de concevoir de façon contrôlée des extensions du système GCC-MM. Nous commençons avec un exemple concret d’application d’une telle démarche, l’introduction des règles **D** (sur la base de [14]), un ensemble de règles permettant d’étendre l’associativité de GCC, et pour lequel la base CTL associée permet d’attribuer les modalités de façon cohérente avec le reste du système, ainsi que de calculer leur effet sur les structures prédicatives construites. Nous ouvrons ensuite une discussion concernant la signification et les intérêts d’une telle démarche, en tentant de préciser la nature des liens entre GCC-MM et CTL qu’elle exploite. Puis, nous terminons par une récapitulation des analogies qu’on peut établir entre ces deux formalismes, afin de mener à bien cette méthode.

### 5.1 L’exemple du combinateur **D**

Note : cette sous-partie emploie le système de modalités de [2]  $M_{MM-GCC} = \{\star, \diamond, \times, \cdot\}$  (version simplifiée sans directionnalités), qui a été étudié en détail lors de la partie précédente.

Nous avons justifié dans la partie 1 l’intérêt de pouvoir mettre en place une stratégie d’analyse incrémentale, et vu comment, en GCC, les règles de levée de type et de composition harmonique introduisent un certain degré d’associativité, qui rend possible l’incrémentalité de l’analyse pour une importante classe de phrases. Cependant, pour certaines constructions, ce degré d’associativité s’avère insuffisant. Prenons par exemple la phrase : *Le livre que je lis est passionnant.*

Jusqu’aux trois premiers mots, tout va bien, les fragments *Le livre* et *Le livre que* peuvent tous deux être analysés comme des constituants :

$$\frac{\frac{\text{Le}}{\text{NP/N}} \quad \frac{\frac{\text{livre}}{\text{N}}}{\text{N/(N\N)}} > \mathbf{T} \quad \frac{\text{que}}{\text{N\N/(S/NP)}}}{\text{NP/(S/NP)}} > \mathbf{B}$$

Cependant, le système ne dispose pas d’une associativité suffisante pour combiner directement *Le livre que* et *je*, et analyser *Le livre que je* comme un opérateur prenant un verbe transitif ( $S \backslash NP / NP$ ) sur sa droite pour former un groupe nominal ( $NP / ((S \backslash NP / NP))$ ), il n’y a pas d’autre recours que d’« attendre » l’intervention du verbe, lequel peut se combiner avec *je* par levée de type ( $je : NP \rightarrow S / (S \backslash NP)$ ) puis composition harmonique, formant un opérateur de type  $S / NP$  consommable par *Le livre que*.

On peut, bien sûr, anticiper cette étape en effectuant immédiatement la levée de type de *je*, mais la composition n’est toujours pas possible :

$$\frac{\frac{\text{Le livre que}}{\text{NP/(S/NP)}} \quad \frac{\text{je}}{\text{NP}}}{\text{S/(S\NP)}} > \mathbf{T}$$

\*\*\* ?

Cependant, tous les arguments présents dans le type cible  $\text{NP}/(\text{S}\backslash\text{NP}/\text{NP})$  sont désormais présents dans les prémisses. En fait, on remarque que l'analyse pourrait continuer si nous disposions d'une règle combinatoire comme :

$$>\mathbf{D} : \text{X}/(\text{Y}/\text{Z}) \text{Y}/\text{W} \rightarrow \text{X}/(\text{W}/\text{Z})$$

Nous reprenons la notation de [14] qui désigne les règles de ce type par la lettre  $\mathbf{D}$ . En fait, une telle règle peut effectivement être dérivée comme un théorème du formalisme CTL, à partir du noyau logique et de la règle structurelle d'associativité. L'observation de la preuve va nous permettre d'une part de repérer où interviennent les règles structurelles d'associativité, et donc d'associer les modalités de façon cohérente avec le reste du système (nous introduirons directement les « bonnes » modalités dans les prémisses de la preuve, ce choix s'expliquera au cours du calcul par l'intervention des règles structurelles les nécessitant). D'autre part, nous allons pouvoir calculer la structure prédicative résultant de l'application de cette nouvelle règle (i.e. son combinateur associé) :

$$\frac{\frac{\frac{\Gamma \vdash u : \text{X}/\diamond(\text{Y}/\diamond\text{Z})}{\Gamma \circ_\diamond (\Delta \circ_\diamond f) \vdash u(\lambda x.v(w(x))) : \text{X}} \text{MLA}}{(\Gamma \circ_\diamond \Delta) \circ_\diamond f \vdash u(\lambda x.v(w(x))) : \text{X}} \text{MLA}}{\Gamma \circ_\diamond \Delta \vdash \lambda w.u(\lambda x.v(w(x))) : \text{X}/\diamond(\text{W}/\diamond\text{Z})} \text{/}\diamond\text{I}}{\frac{\frac{\frac{\frac{\Delta \vdash v : \text{Y}/\diamond\text{W}}{\Delta \circ_\diamond (f \circ_\diamond x) \vdash v(w(x)) : \text{Y}} \text{MLA}}{(\Delta \circ_\diamond f) \circ_\diamond x \vdash v(w(x)) : \text{Y}} \text{/}\diamond\text{I}}{\Delta \circ_\diamond f \vdash \lambda x.v(w(x)) : \text{Y}/\diamond\text{Z}} \text{/}\diamond\text{E}}{\frac{\frac{f \vdash w : \text{W}/\diamond\text{Z} \quad x \vdash z : \text{Z}}{f \circ_\diamond x \vdash w(z) : \text{W}} \text{/}\diamond\text{E}}{\Delta \circ_\diamond (f \circ_\diamond x) \vdash v(w(x)) : \text{Y}} \text{/}\diamond\text{E}}{\Delta \vdash v : \text{Y}/\diamond\text{W}} \text{/}\diamond\text{E}}{\Gamma \vdash u : \text{X}/\diamond(\text{Y}/\diamond\text{Z})} \text{/}\diamond\text{E}}{\Gamma \circ_\diamond \Delta \vdash \lambda w.u(\lambda x.v(w(x))) : \text{X}/\diamond(\text{W}/\diamond\text{Z})} \text{/}\diamond\text{I}}$$

(dans notre exemple,  $\Delta$  était *le livre que*,  $\Gamma$  était *je* après levée de type, l'élément hypothétique  $f$  peut être interprété comme une référence au verbe transitif, *lis*, qui n'a pas encore été introduit, et  $z$  correspond à l'objet du verbe -le livre- qui est extrait).

On constate donc que 5.1 est bien un théorème CTL, et nécessite deux fois l'intervention de la règle structurelle d'associativité, qui impose à tous les slashes des prémisses de porter la modalité  $\diamond$ . On pourrait également relever le fait qu'elle n'introduit aucun effet permutatif (donc n'altère pas l'ordre des mots, et ne nécessite pas l'intervention de la modalité  $\times$ ). On peut ainsi déduire la règle GCC-MM :

$$>\mathbf{D} : \text{X}/\diamond(\text{Y}/\diamond\text{Z}) \text{Y}/\diamond\text{W} \rightarrow \text{X}/\diamond(\text{W}/\diamond\text{Z})$$

De façon intéressante, les étapes de la preuve que nous donnons ici correspondent (hormis les deux dernières) à l'analyse non-incrémentale de la phrase que nous avons évoquée plus haut<sup>17</sup>, en interprétant l'hypothèse  $f$  de façon naturelle comme le verbe transitif non encore introduit, qui se combine d'abord avec son sujet avant d'être consommé par la construction relative. On peut ainsi considérer, de façon intuitive, qu'appliquer la règle  $\mathbf{D}$  de GCC-MM implique une hypothèse sur les éléments qui vont suivre<sup>18</sup> (c'est en ce sens que [33] utilise le terme *predictive combinator* pour désigner de telles règles), intuition qui, dans la preuve CTL de  $\mathbf{D}$ , se traduit formellement par l'utilisation du mécanisme de l'introduction d'hypothèses.

On note également que la preuve CTL nous offre un moyen calculatoire efficace pour déterminer comment une règle combine les structures prédicatives de ses prémisses en une structure résultante. Ainsi, le calcul précédent nous donne :

$$\mathbf{D}w \rightarrow \lambda w.u(\lambda x.v(w(x)))$$

On peut transformer cette lambda-expression en expression combinatoire, par la transformation usuelle  $\mathbf{T}[\ ]$  :

17. Raison pour laquelle nous avons préféré présenter cette preuve à celle donnée dans [14], où ceci n'était pas visible de façon directe.

18. Quoique la deuxième hypothèse  $z$ , ne peut pas, elle, être interprétée intuitivement de la même manière exactement, car elle ne sera jamais instanciée effectivement, elle correspond en fait à la place qu'occuperait l'objet extrait, ici le livre, dans la construction canonique induite par la catégorie du verbe. Mais on pourrait arguer que l'hypothèse  $z$  est en quelque sorte subordonnée à l'hypothèse  $w$ , et pousser jusqu'au bout l'interprétation intuitive en disant que l'application de la règle entre *Le livre que* et *je* fait l'hypothèse que ces éléments seront suivis par un verbe qui lui-même fait l'hypothèse (naïve) d'être suivi par son objet (il ne « sait » pas qu'il appartient à une construction relative, étant en position d'opérande par rapport à cette dernière).

$$\begin{aligned}
T[\lambda w.u(\lambda x.v(w(x)))] &= \mathbf{B} u T[\lambda w.\lambda x.v(w(x))] \\
&= \mathbf{B} u T[\lambda w.T[\lambda x.v(w(x))]] \\
&= \mathbf{B} u T[\lambda w.T[\lambda x.v(w(x))]] \\
&= \mathbf{B} u T[\lambda w.T[\lambda x.v(w(x))]] \\
&= \mathbf{B} u T[\lambda w.(\mathbf{B} v w)] && (\eta\text{-réduction}) \\
&= \mathbf{B} u (\mathbf{B} v)
\end{aligned}$$

On remarque qu'on peut définir  $\mathbf{D}$  par un schéma de  $\beta$ -réduction à deux opérandes, consistant à appliquer le combinateur  $\mathbf{B}$  à chacune des opérandes ( $D u v \xrightarrow{\beta} \mathbf{B} u (\mathbf{B} v)$ ).

On peut si, on le désire, poursuivre le calcul afin d'exprimer  $\mathbf{D}$  intrinsèquement comme combinaison de combinateurs plus élémentaires :

$$\begin{aligned}
T[\lambda u.\lambda v.\mathbf{B} u (\mathbf{B} v)] &= T[\lambda u.T[\lambda v.\mathbf{B} u (\mathbf{B} v)]] \\
&= T[\lambda u.T[\lambda v.\mathbf{B} u (\mathbf{B} v)]] \\
&= T[\lambda u.\mathbf{B} (\mathbf{B} u) \mathbf{B}] \\
&= \mathbf{C} T[\lambda u.\mathbf{B} (\mathbf{B} u)]\mathbf{B} \\
&= \mathbf{C} (\mathbf{B} \mathbf{B} \mathbf{B}) \mathbf{B}
\end{aligned}$$

Ainsi, puisque par définition<sup>19</sup>  $\mathbf{B} \mathbf{B} \mathbf{B} = \mathbf{B}^2$ , on a  $\mathbf{D} = \mathbf{C} \mathbf{B}^2 \mathbf{B}$

On peut vérifier que ce résultat est bien cohérent avec les structures prédicatives formées en utilisant les anciennes règles (on posera  $J =_{def} \mathbf{B}$  le (( $\mathbf{C}_*$  journal) que), et on admettra que  $J$  est la structure prédicative sous-jacente au fragment *Le journal que*)

1. Sans la règle  $\mathbf{D}$  :

$$\frac{\frac{\text{Le journal que}}{J : \text{NP}/(\text{S}/\text{NP})} \quad \frac{\frac{\text{je}}{\mathbf{je} : \text{NP}}}{\mathbf{C}_* \mathbf{je} : \text{S}/(\text{S}\backslash\text{NP})} > \mathbf{T} \quad \frac{\text{lis}}{\mathbf{lis} : \text{S}\backslash\text{NP}/\text{NP}}}{\frac{\mathbf{B} (\mathbf{C}_* \mathbf{je}) \mathbf{lis} : \text{S}/\text{NP}}{J (\mathbf{B} (\mathbf{C}_* \mathbf{je}) \mathbf{lis}) : \text{S}}} > \mathbf{B}$$

2. Avec la règle  $\mathbf{D}$  :

$$\frac{\frac{\text{Le journal que}}{J : \text{NP}/(\text{S}/\text{NP})} \quad \frac{\frac{\text{je}}{\mathbf{je} : \text{NP}}}{\mathbf{C}_* \mathbf{je} : \text{S}/(\text{S}\backslash\text{NP})} > \mathbf{T}}{\mathbf{D} J (\mathbf{C}_* \mathbf{je}) : \text{S}/(\text{S}\backslash\text{NP}/\text{NP})} > \mathbf{D} \quad \frac{\text{lis}}{\mathbf{lis} : \text{S}\backslash\text{NP}/\text{NP}}}{\mathbf{D} J (\mathbf{C}_* \mathbf{je}) \mathbf{lis} : \text{S}} >$$

$$\begin{aligned}
\mathbf{D} J (\mathbf{C}_* \mathbf{je}) \mathbf{lis} &= \mathbf{C} \mathbf{B}^2 \mathbf{B} J (\mathbf{C}_* \mathbf{je}) \mathbf{lis} \\
&= \mathbf{B}^2 J \mathbf{B} (\mathbf{C}_* \mathbf{je}) \mathbf{lis} \\
&= J (\mathbf{B} (\mathbf{C}_* \mathbf{je}) \mathbf{lis})
\end{aligned}$$

La dérivation utilisant la règle  $\mathbf{D}$  (et le combinateur associé que nous avons établi) débouche donc bien sur la même structure prédicative que la dérivation « classique ».

[14] étudie plusieurs autres règles de forme similaire à  $\mathbf{D}$ , par exemple

$$> \mathbf{D}_{\diamond \times} : X /_{\diamond} (Y \backslash_{\times} Z) Y /. W \rightarrow X /. (W \backslash_{\times} Z)$$

19. La puissance nième d'un combinateur  $\chi$  étant définie par  $\chi^1 =_{def} \chi$  et  $\chi^n =_{def} \mathbf{B} \chi \chi^{n-1}$

Dont nous donnons ci-dessous la preuve CTL associée (nous n'avons pas recopié le calcul de la structure prédicative, celui-ci étant strictement identique à celui de la démonstration précédente -les règles structurelles étant supposées sémantiquement inertes-, comme le lecteur pourra le vérifier) :

$$\begin{array}{c}
\frac{\Delta \vdash Y/W \quad \frac{x \vdash Z \quad f \vdash W \backslash_{\times} Z}{x \circ_{\times} f \vdash W} \backslash_{\times} E}{\Delta \circ (x \circ_{\times} f) \vdash Y} /_{\diamond} E \\
\frac{\Delta \circ (x \circ_{\times} f) \vdash Y}{x \circ_{\times} (\Delta \circ f) \vdash Y} \text{MLP} \\
\frac{\Gamma \vdash X /_{\diamond} (Y \backslash_{\times} Z) \quad \frac{\Delta \circ f \vdash Y \backslash_{\times} Z}{\Delta \circ f \vdash Y \backslash_{\times} Z} \backslash_{\times} I}{\Gamma \circ_{\diamond} (\Delta \circ f) \vdash X} /_{\diamond} E \\
\frac{\Gamma \circ_{\diamond} (\Delta \circ f) \vdash X}{(\Gamma \circ_{\diamond} \Delta) \circ f \vdash X} \text{MLA} \\
\frac{(\Gamma \circ_{\diamond} \Delta) \circ f \vdash X}{\Gamma \circ_{\diamond} \Delta \vdash X /_{\diamond} (W \backslash_{\times} Z)} /_{\diamond} I
\end{array}$$

On en déduit l'attribution des modalités adéquates aux prémisses et conclusion de la règle  $>D_{\diamond \times}$  : les deux slashes de la catégorie de  $\Gamma (X /_{\diamond} (Y \backslash_{\times} Z))$  peuvent se contenter de porter respectivement les modalités  $\diamond$  et  $\times$  tandis que le produit formé par le slash de  $\Delta$  est impliqué à la fois dans une règle structurelle permutative et une règle associative : il lui faut porter la modalité générale  $\ll \cdot \gg$ . La conclusion, quant à elle, doit être de type  $X /_{\diamond} (W \backslash_{\times} Z)$ <sup>20</sup>.

L'exemple des règles **D** montre ainsi combien l'interaction entre CTL et GCC-MM peut être fructueuse pour la conception de nouvelles règles. L'utilisation nous permet de révéler les différents effets induits par les règles (association, permutation...), et de déterminer les modalités à utiliser pour maintenir la cohérence avec le reste du système. En outre, la preuve CTL sous-jacente à une règle GCC-MM fournit également la structure prédicative résultant de l'application de la règle.

## 5.2 Discussion : liens entre CTL, GCC-MM, GCCA

On a vu se dessiner dans les parties précédentes les grandes lignes de ce que pourrait être une utilisation conjointe de GCC-MM et CTL, et les avantages d'une telle combinaison. La meilleure expressivité de CTL permet dans un premier temps de modéliser bien plus facilement les phénomènes linguistiques que l'on veut couvrir, les mécanismes d'introduction d'hypothèses, de règles structurelles contrôlées par des modalités offrant un langage riche, souple et robuste (rendant le contrôle de la surgénération facile), ce en quoi il est bien adapté à cette tâche d'intégration des données linguistiques (cf. par exemple les preuves CTL associées aux règles croisées, l'analyse des clitiques présentée en 2.3). CTL permet ainsi de définir le  $\ll$  comportement  $\gg$  que l'on veut obtenir d'un système, en particulier les phrases qu'il rejettera/ acceptera.

Cette étape ayant été effectuée, la suite de la démarche générale que nous avons dépeinte (à la suite de [2]) consiste à *compiler* les preuves CTL qui constituent notre analyse en règles combinatoires GCC-MM. GCC-MM ne dispose pas des mécanismes d'introduction d'hypothèses, ni de règles structurelles, ces mécanismes CTL doivent être considérés comme implicites (compilés) dans les règles GCC-MM qu'elles sous-tendent. L'intérêt d'une telle  $\ll$  compilation  $\gg$  réside bien sûr dans le gain substantiel en terme de puissance algorithmique requise. On peut ainsi voir les règles GCC-MM comme des raccourcis à certaines preuves CTL, permettant de condenser les différentes étapes qui séparent les prémisses de la conclusion en une seule, sans introductions d'hypothèses ou manipulations structurelles explicites. Cette démarche peut aboutir, typiquement, à l'introduction de nouvelles règles, permettant d'exporter dans GCC-MM certains comportements disponibles dans CTL. L'exemple des règles **D** est en ce sens assez éloquent, en permettant à GCC-MM de disposer d'un plus grand degré d'associativité, tout en maintenant la cohérence du système de modalités et évitant la surgénération. En effet, supposons que le système CTL sur lequel on se base (construit à la première étape) n'entraîne pas de surgénération. Puisque tout calcul GCC-MM peut être analysé comme un raccourci d'une preuve CTL (c'est à dire que tout calcul GCC-MM *est* en fait une preuve CTL), on a pour conséquence immédiate que la **propriété de non-surgénération est conservée** dans la transposition de CTL à GCC-MM.

20. [14] donne le type  $X /_{\diamond} (W \backslash_{\times} Z)$ , c'est une erreur, la modalité que doit porter le slash principal est bien  $\cdot$  (comme on peut le voir facilement dans la preuve ci-dessus).

La comparaison entre les règles **B** et **D** est d'ailleurs intéressante, car elle éclaire sur la nature de la relation entre les règles GCC-MM et les constituants (en particulier, les règles structurelles) de leurs preuves CTL sous-jacentes. Nous avons vu en effet que plusieurs d'entre elles pouvaient être vues comme la traduction dans GCC d'une règle structurelle CTL, par exemple la règle  $\triangleright\mathbf{B}$  comme une traduction de la règle d'associativité à gauche, de même que la règle **D** (voir parties précédentes).

Cependant, ni la règle **B**, ni la règle **D** ne sont équivalentes à l'action combinée de la règle structurelle d'associativité et de la base logique de CTL (sinon, une seule serait nécessaire...). Le formalisme GCC-MM, de part le fait même qu'il s'interdit les mécanismes d'introduction d'hypothèses et de règles structurelles pour conserver (entre autres) ses bonnes propriétés algorithmiques, est condamné (évidemment) à rester incomplet par rapport à CTL : même en raffinant successivement un système GCC-MM, il demeurera toujours des preuves CTL qui ne sont pas transposables en un calcul GCC-MM (sinon les deux systèmes auraient la même puissance générative). La puissance de la base logique de CTL joue un rôle important dans cette différence. On pourra remarquer que la preuve de la règle **D** fait appel à deux introductions d'hypothèses, tandis que **B** n'en nécessite qu'une. Ce qui différencie la preuve de ces deux règles en CTL est donc le degré sollicitation de la base logique, dont chacune des deux règles ne capture qu'une (petite) partie.

Toute tentative de conversion d'une analyse CTL en GCC-MM (comme celle que nous présentons dans la dernière partie de ce rapport) doit tenir compte de cette limitation intrinsèque. On peut distinguer deux leviers principaux dont dispose GCC pour repousser les limites (i.e. les sous-génération) découlant de cette « incomplétude » :

- **L'introduction d'une nouvelle règle** (cf. les règles **D**), que nous venons de discuter. À noter que ces nouvelles règles peuvent simplement consister en de nouvelles versions modales de règles existantes (cf. la discussion sur la modalité  $\times!$  dans la partie 4.3). Bien sûr, l'introduction d'une nouvelle règle doit se faire avec la plus grande circonspection, la philosophie qui motive GCC (et à plus forte raison de GCC-MM) considérant que l'ensemble des règles doit obéir à un certain critère d'universalité. Ceci a l'intérêt d'éviter l'introduction de règles ad hoc à une situation particulière, qui « noient » l'analyse linguistique et rendent le système inextricable).

Il convient de choisir les « bons » théorèmes de CTL à transposer dans GCC-MM, de façon à obtenir un ensemble de règles aussi petit que possible. Noter que l'ensemble de règles varie selon les version de (MM-)GCC ; certaines (comme [32]) n'ont pas recours aux règles croisées et utilisent à la place une règle dite de « permutation faible généralisée » :  $((X|Y_1)...|Y_n) \rightarrow ((X|Y_n)|Y_1...)$ .

On peut éventuellement adopter un point de vue moins strict en distinguant deux types de règles, d'une part les règles combinatoires universelles, d'autre part les règles « lexicales », i.e. des règles unaires ne s'appliquant qu'aux catégories du lexique, servant à générer tous les types d'une entité à partir d'un « type de base » ([28] considère, par exemple, la levée de type comme appartenant à cette dernière catégorie<sup>21</sup>).

- **L'introduction de nouveaux types.** Un moyen simple et classique (et parfois inévitable) de résoudre certaines sous-génération est d'avoir recours à l'*ambiguïté lexicale*, c'est à dire d'attribuer un (des) type(s) supplémentaire(s) à certaines unités linguistiques pour couvrir la ou les constructions visées. Il peut bien évidemment être employé pour pallier les imperfections d'une transposition de CTL vers GCC.

Concernant le levier de l'ambiguïté lexicale, prenons un exemple important. Nous avons vu dans la partie [15] l'utilisation de l'opérateur d'extraction  $\uparrow$  pour définir le type des clitiqes (SV/(SV $\uparrow$ NP), qui traduit l'idée intuitive qu'un clitique prend en argument un verbe à qui il manque un NP « quelque part », pour former un groupe verbal complet). Un tel opérateur ne serait pas directement définissable en GCC ou GCC-MM (sous leur forme actuelle), du fait de l'inexistence d'un mécanisme de manipulation des structures (comme les règles structurelles). Cependant l'ambiguïté lexicale permet de le simuler de façon relativement satisfaisante dans de nombreux cas.

Pour mettre ceci en évidence, étudions d'abord de plus près cet « opérateur d'extraction ». [22] modélise  $\uparrow$  à l'aide d'une modalité unaire, que nous noterons ici  $\diamond_s$ , en posant  $A \uparrow B =_{def} A/\diamond_s \square_s B$  et en lui associant trois règles structurelles permettant d'obtenir le comportement souhaité :

21. Ce qui est assez discutable, puisque par exemple l'analyse incrémentale d'une phrase comme *Le chat boit du lait* impose d'effectuer une levée de type de *Le chat* (NP), qui n'est pas un élément du lexique.

$$\frac{\Gamma\{\Delta_1 \circ (\Delta_2 \circ \Delta_3)\} \vdash X}{\Gamma\{(\Delta_1 \circ \Delta_2) \circ \Delta_3\} \vdash X} \text{A}, \quad \frac{\Gamma\{\Delta_1 \circ \Delta_2\} \vdash X}{\Gamma\{\Delta_2 \circ \Delta_1\} \vdash X} \text{C}, \quad \frac{\Gamma\{(\Delta_1 \circ \Delta_3) \circ \Delta_2\} \vdash X}{\Gamma\{(\Delta_1 \circ \Delta_2) \circ \Delta_3\} \vdash X} \text{C''}$$

Avec l'un des  $\Delta_i$  de la forme  $\langle \Lambda \rangle_s$

Ces règles permettent à une hypothèse marquée par la modalité  $\langle \rangle_s$  de « glisser » vers la périphérie de la structure, d'où elle peut être extraite par les slashes classiques  $/, \backslash$ , et ainsi de réaliser exactement le comportement de l'opérateur d'extraction.

On se souvient que dans 2.3 le mot *en* se voyait attribuer (aux contraintes d'ordonnancement près) le type  $SV/(SV \uparrow NP_{acc})$  (=  $SV/(SV/\diamond_s \square_s^{\downarrow} NP)$  donc), qui permettait de le combiner avec des verbes transitifs (par exemple, *vouloir* :  $S/NP_{acc}$ , *J'en veux*) tout comme ditransitifs (e.g. *donner* :  $S/NP_{dat}/NP_{acc}$ , *Je lui en donne*). Une telle concision n'est pas possible en GCC. En effet, pour le dernier cas la position de  $NP_{acc}$  dans la catégorie du verbe, inaccessible aux règles de composition, impose d'introduire pour *en* le type supplémentaire  $(SV/NP_{dat})/(SV/NP_{dat}/NP_{acc})$  (ou bien de faire appel à une règle combinatoire supplémentaire).

Évidemment le type CTL se rapproche plus de ce qu'« est » conceptuellement un clitique objet comme *en* dans toute sa généralité (un opérateur qui extrait un argument  $NP_{acc}$  *quelque part*), tandis que les deux types de GCC se bornent à énumérer son action dans plusieurs situations distinctes (devant un verbe transitif, devant un verbe ditransitif...) sans réussir à isoler l'invariant qu'elles ont en commun, et donc à les abstraire en un type unique. Cependant, cette stratégie de recours à l'ambiguïté lexicale a le mérite d'être rapide à mettre en place, et de pallier facilement certaines insuffisances du système, ce qui dans un contexte applicatif est hautement appréciable.

En outre, ce procédé vérifie aussi la propriété de conservation de la non-surgénération de CTL vers GCC-MM. En effet, il suffit de montrer que le système CTL sous-jacent au système GCC-MM utilisé dérive le type supplémentaire, et alors la conservation est assurée. Le nouveau type peut donc être lui aussi vu comme un raccourci, une « compilation » d'une certaine preuve CTL. Comme dans le cas des règles, ceci peut donc également aider à attribuer les différentes modalités du nouveau type de façon cohérente avec le reste du système. Par exemple, dans le cas de *en*, la preuve CTL ci-dessous permet de dériver le type supplémentaire introduit dans GCC-MM (l'hypothèse  $v$  peut être analysée comme la prévision d'une combinaison avec un verbe ditransitif) :

$$\frac{\frac{\frac{\frac{x_a \vdash \square_s^{\downarrow} NP_{acc}}{\langle x_a \rangle_s \vdash NP_{acc}} \square_s^{\downarrow} E}{v \vdash SV/NP_{dat}/NP_{acc}}{v \circ \langle x_a \rangle_s \vdash SV/NP_{dat}} /E}{\frac{(v \circ \langle x_a \rangle_s) \circ x_d \vdash SV}{(v \circ x_d) \circ \langle x_a \rangle_s \vdash SV} \text{C''}}{x_d \vdash NP_{dat}} /E}{\frac{(v \circ x_d) \circ x'_a \vdash SV}{v \circ x_d \vdash SV/\diamond_s \square_s^{\downarrow} NP_{acc}} /I} \diamond_s E}{\frac{\frac{\frac{\frac{en \vdash SV/(SV/\diamond_s \square_s^{\downarrow} NP_{acc})}{en \circ (v \circ x_d) \vdash SV} \text{MA}}{(en \circ v) \circ x_d \vdash SV} /I}{en \circ v \vdash SV/NP_{dat}} /I}{en \vdash SV/NP_{dat}/(SV/NP_{dat}/NP_{acc})} /I} /E$$

L'introduction d'un type supplémentaire peut donc elle aussi disposer d'un appui CTL de façon à prévenir la surgénération.

Deux autres leviers sont envisageables pour pallier l'incomplétude de GCC par rapport à CTL. Le premier est une exploitation plus avancée des relations d'héritage (comme par exemple dans [20] ou [6]). La structuration de l'ensemble des types et de leurs traits, en introduisant des types génériques, permet en effet de supprimer les redondances du lexique et d'abstraire certaines situations en remplaçant plusieurs types par un seul (le type générique décrivant le concept (l'« intension ») de l'ensemble qui, auparavant, était énuméré par les types multiples (l'« extension » de l'ensemble-cible en question). En outre, de telles structures pourraient permettre d'interpréter certaines modalités unaires CTL, et règles structurelles comme des traits<sup>22</sup> reliés par certaines

22. Noter cependant qu'il n'est pas immédiat de passer d'une modalité CTL à un trait GCC(-MM), les modalités CTL pouvant s'appliquer à des types complexes tandis qu'en GCC, généralement, seuls les types atomiques peuvent porter de traits (ajouter un trait équivaut alors à ajouter un type atomique).

relations d'héritage ([14]), évitant le recours à l'introduction de nouvelles règles ou à l'ambiguïté lexicale.

Il importe cependant de contrôler que les mécanismes d'héritage utilisés ne conduisent pas à une augmentation de la puissance générative par laquelle le système perdrait ses propriétés algorithmiques attractives.

Le second est de lier certains mécanismes de contrôle, notamment les métarègles de [7], à une interprétation CTL. On a vu que la levée de type pouvait être analysée comme l'introduction de l'hypothèse d'une combinaison future avec un certain opérateur (qui apparaît explicitement dans la preuve CTL correspondante). L'action des métarègles consiste, sous cette interprétation, à chercher dans le contexte des « bonnes raisons » d'invoquer telle ou telle hypothèse<sup>23</sup>, donc pour concevoir de telles métarègles, il est utile d'être conscient des hypothèses implicites dans l'utilisation de telle ou telle règle. Il serait intéressant de comparer ces métarègles aux heuristiques de filtrage utilisées par [24] pour l'analyse syntaxique effective de grammaires basées sur CTL, à la base du système *Grail* (<http://www.labri.fr/perso/moot/grail3.html>). Il serait également intéressant, soit dit en passant, de chercher à transposer certains mécanismes avancés de GCCA, telles les règles de décomposition/réorganisation structurelle, au sein de CTL, et éventuellement de les lier à un système de modalités permettant de se prémunir des surgénérations.

### 5.3 Récapitulation des analogies entre GCC-MM et CTL et GCCA

En conclusion de cette partie, nous résumons dans un tableau comparatif les correspondances et analogies que nous avons pu dégager entre les formalismes GCC-MM, GCCA, et CTL. Les cases comportant un point d'interrogation dénotent des analogies non explorées en profondeur dans ce rapport, qui peuvent être considérées comme des pistes de travaux futurs.

CTL	GCC-MM	GCCA
Base logique fixe (règles générales d'introduction et élimination) + règles structurelles stipulées	Petit ensemble de règles combinatoires, recours plus important à l'ambiguïté lexicale	
<b>Théorèmes</b> du système	Règles combinatoires <b>stipulées</b>	
Modalités binaires	Modalités de slash (comme dans [2])	(non utilisées)
Modalités unaires	Traits portés par les catégories atomiques (à étendre aux catégories complexes ?)	
Interactions entre les modalités au sein des règles structurelles	Relations d'héritage entre traits et modalités ?	
Manipulation des structures construites via les <b>règles structurelles</b>	Aucune, implicites dans l'application des règles combinatoires, lesquelles ne peuvent que <i>construire</i> les structures	Règles de décomposition et réorganisation structurelle ?
Introduction d'hypothèses	Aucune, également implicites dans l'application des règles (notamment de levée de type)	
Heuristiques de filtrage pour l'analyse syntaxique	(non explicites)	Métarègles ?

23. L'une des métarègles de [7] stipule par exemple : « Si on est en présence d'une suite NP - X\NP/Y, alors appliquer au premier NP la levée de type X/(X\NP) ». Dans le cas où X=S, Y=NP, cela revient à observer que le mot suivant le NP est un verbe transitif, et donc d'appliquer en conséquence une levée de type intégrant l'hypothèse d'une consommation du NP par un verbe sur la gauche de celui-ci.

## 6 Application : Conception d'un système MM-CCG pour le traitement des clitiques du français

Dans cette partie, nous tentons (modestement) d'appliquer la méthode de conception des règles GCC sur la base des CTL proposée par [2], et étudiée dans les deux parties précédentes, au traitement des clitiques du français, sur la base de l'analyse CTL de [15]

### 6.1 Élaboration des règles MM-CCG compilant l'analyse de Kraak

En tentant de transposer une analyse CTL en GCC-MM, deux difficultés principales se présentent. D'une part, comme nous l'avons signalé précédemment, le système de modalités unaires de CTL n'est pas équivalent au système de traits habituellement utilisé dans GCC-MM, ces derniers ne s'appliquant qu'aux catégories *atomiques* ([2]), constituant ainsi une simple extension de l'ensemble de celles-ci, qui reste fini. Les modalités unaires CTL, quant à elles, peuvent s'appliquer à des catégories complexes (une catégorie comme  $\diamond(S \setminus NP/NP/NP)$  est permise par exemple). En outre, dans toute leur généralité, les modalités CTL peuvent s'appliquer sans limite sur une entité ( $\diamond\diamond\dots\diamond\diamond\diamond(A)$ ) en formant à chaque fois un objet différent, ce qui n'est pas possible en GCC car l'ensemble  $\mathcal{T}_0$  des types atomiques doit être fini.

Pour les besoins de la transposition, nous ferons donc l'hypothèse que les catégories de GCC-MM peuvent être elles aussi encapsulées dans des opérateurs  $\diamond_i, \square_i^\downarrow$ , l'ensemble  $\mathcal{T}$  des types de GCC-MM étant alors défini par  $\mathcal{T} := \mathcal{T}_0 | \square_i^\downarrow \mathcal{T} | \diamond_i \mathcal{T} | \mathcal{T} /_j \mathcal{T} | \mathcal{T} \setminus_j \mathcal{T}$  avec  $i \in \mathcal{M}_{unaires}$  et  $j \in \mathcal{M}_{binaires}$ .

Cette hypothèse est parfois utilisée dans la littérature, par exemple dans [28] (il serait intéressant, dans des travaux ultérieurs, de déterminer précisément s'il est possible ou non de se passer de cette hypothèse). Notons qu'elle n'introduit pas de différence notable dans la façon de réaliser le calcul, il suffit d'ajouter à la condition d'égalité entre deux catégories une condition comme «  $\diamond_i A$  s'unifie à  $B$  ssi  $B$  est de la forme  $\diamond_i C$  avec  $C$  s'unifiant à  $A$  » (de même pour  $\square_i^\downarrow$ ). Les modalités binaires, elles, ne posent pas de problème dans le cadre de GCC-MM : il suffit d'utiliser autant de décoration de slash (pour les deux sens) qu'il existe de modalités CTL dans le système à transposer. Nous introduisons donc ici les slashes :  $\{ /_c, \setminus_c, /_{ca}, \setminus_{ca}, /_u, \setminus_u \}$ . Nous supposons en outre que les slashes  $/_{ca}, \setminus_{ca}$  ne dérivent pas  $\times$ , et donc sont impropres à intervenir dans des règles croisées (ce qui est logique au vu des contraintes strictes d'ordonnancement auxquelles ils sont liés).

La deuxième difficulté, plus fondamentale encore, concerne l'inexistence en GCC-MM de mécanismes comparables aux règles structurelles (autrement dit, GCC n'a accès qu'au « côté droit » des preuves CTL). L'analyse de Kraak en utilisait 7 : K1, K2, K,  $\emptyset$ , I,  $K_u$  et MA. Le but est d'intégrer dans GCC des règles modales (dérivées comme des théorèmes de CTL), capturant les effets de ces règles structurelles.

Intéressons nous pour commencer à la règle K :

$$\frac{\Gamma\{\langle \Delta \rangle_l^y \circ_{ca} \langle \Theta \rangle_l^z\} \vdash A}{\Gamma\{\langle \Delta \circ_{ca} \Theta \rangle_l^x\} \vdash A} \text{K} \quad x \preceq y \prec z$$

Notre problème en GCC est que nous n'avons pas accès à la structure  $\langle \Delta \rangle_l^y \circ_{ca} \langle \Theta \rangle_l^z$ . Cependant, observons que la seule façon en CTL de former un tel produit  $\Delta_1 \circ \Delta_2$  est d'invoquer (dans les versions n'utilisant pas le produit  $\Delta_1 \bullet \Delta_2$ ) les règles d'élimination de slash  $/E$  et  $\setminus E$ <sup>24</sup>, lesquelles impliquent l'intervention d'un opérateur  $A/B$ , à l'instar des règles d'application de CGG. Une idée serait donc de capturer l'action des règles structurelles par une action sur la catégorie de l'opérateur qui serait le miroir de l'action de la règle K.

$$\frac{\frac{\frac{\Delta \vdash \square_l^{\downarrow y} (A /_{ca} B)}{\langle \Delta \rangle_l^y \vdash A /_{ca} B} \square_l^{\downarrow y} E \quad \frac{u \vdash \square_l^{\downarrow z} B}{\langle u \rangle_l^z \vdash B} \square_l^{\downarrow z} E}{\frac{\langle \Delta \rangle_l^y \circ_{ca} \langle u \rangle_l^z \vdash A}{\langle \Delta \circ_{ca} u \rangle_l^x \vdash A} /_{ca} E} \text{K } x \preceq y \prec z}{\frac{\langle \Delta \circ_{ca} u \rangle_l^x \vdash A}{\Delta \circ_{ca} u \vdash \square_l^{\downarrow x} A} \square_l^{\downarrow x} I} /_{ca} I}{\Delta \vdash \square_l^{\downarrow x} A /_{ca} \square_l^{\downarrow z} B} /_{ca} I$$

24. En se restreignant à la base logique. Les catégories  $\Delta_1$  et  $\Delta_2$  peuvent en revanche être associées à ce produit après coup, après invocation d'une règle d'associativité ou de permutation par exemple.



On peut ainsi stipuler une règle GCC unaire cohérente avec la base CTL :

$$\mathbf{K}_{GCC-MM-I} : \Box_l^{\downarrow y}(A/caB) \rightarrow \Box_l^{\downarrow x}A/ca\Box_l^{\downarrow z}B$$

avec  $x, y, z$  tels que  $x \preceq y \prec z$

Remarquons qu'à la place de  $\Box_l^{\downarrow y}(A/caB)$ , nous aurions pu mettre  $\Box_l^{\downarrow y}(A/caB)\backslash C/D$  dans la première prémisses, le principe de la preuve étant toujours applicable :

$$\frac{\frac{\frac{\Delta \vdash \Box_l^{\downarrow y}(A/caB)\backslash C/D \quad w \vdash D}{\Delta \circ w \vdash \Box_l^{\downarrow y}(A/caB)\backslash C} /E \quad v \vdash C}{v \circ (\Delta \circ w) \vdash \Box_l^{\downarrow y}(A/caB)} \backslash E \quad \frac{u \vdash \Box_l^{\downarrow z}B}{\langle u \rangle_l^z \vdash B} \Box_l^{\downarrow z}E}{\frac{\langle v \circ (\Delta \circ w) \rangle_l^y \vdash A/caB}{\langle v \circ (\Delta \circ w) \rangle_l^y \circ_{ca} \langle u \rangle_l^z \vdash A} \Box_l^{\downarrow y}E \quad \mathbf{K} \ x \preceq y \prec z}{\frac{\langle (v \circ (\Delta \circ w)) \circ_{ca} u \rangle_l^x \vdash A}{(v \circ (\Delta \circ w)) \circ_{ca} u \vdash \Box_l^{\downarrow x}A} \Box_l^{\downarrow x}I}{v \circ (\Delta \circ w) \vdash \Box_l^{\downarrow x}A/ca\Box_l^{\downarrow z}B} /I} /E$$

$$\frac{\Delta \circ w \vdash \Box_l^{\downarrow x}A/ca\Box_l^{\downarrow z}B\backslash C}{\Delta \vdash \Box_l^{\downarrow x}A/ca\Box_l^{\downarrow z}B\backslash C/D} /E$$

Par suite, on voit aisément que ce résultat s'étend pour  $n$  arguments de direction quelconque. On peut ainsi écrire une version plus générale de la règle ci-dessus :

$$\mathbf{K}_{GCC-MM-I'} : (\Box_l^{\downarrow y}(A/caB))\$ \rightarrow (\Box_l^{\downarrow x}A/ca\Box_l^{\downarrow z}B)\$$$

avec  $x, y, z$  tels que  $x \preceq y \prec z$

(rappel :  $A\$$  correspond à l'ensemble des catégories définissables qui peuvent être interprétées comme un opérateur de résultat  $A$ ,  $A\$ = \{A/X, A\backslash X, A/X/Y, A/X\backslash Y\} \dots$ )

Cette règle nous permet donc d'avoir accès indirectement à l'effet de la règle  $\mathbf{K}$  (vérification de la condition d'ordonnancement), ce qui nous permettra de transposer un certain nombre d'analyses impliquant la règle  $\mathbf{K}$ . Mais ce n'est pas suffisant, et certaines situations demeurent impossibles à transposer (notamment si l'on se restreint à une analyse incrémentale), ce à cause de l'incomplétude signalée au paragraphe précédent. Il nous est ainsi nécessaire de modifier la règle  $\mathbf{D}$  pour y intégrer l'effet de  $\mathbf{K}$  sur les éventuelles modalités portées par les prémisses<sup>25</sup>. On peut par exemple utiliser la règle suivante :

$$\triangleright \mathbf{D}_K : X/ca\Box_l^{\downarrow y}(Y/cZ) \Box_l^{\downarrow y}Y/j \ W \rightarrow X/ca\Box_l^{\downarrow z}(W/cZ)$$

avec  $y, z$  tels que  $y \prec z$

Le lecteur intéressé pourra vérifier que cette règle est bien un théorème de la base CTL en adaptant de la preuve de la règle  $\mathbf{D}$  donnée dans la partie 5.1 (donc n'entraîne pas de surgénération supplémentaire). Il serait bien sûr avantageux, si possible, d'unifier les règles  $\mathbf{K}_{GCC-MM}$  et  $\mathbf{D}_K$  via un seul mécanisme.

On peut de façon symétrique dériver la règle  $\mathbf{K}$  suivante pour les opérateurs arrières :

$$\mathbf{K}_{GCC-MM-II} : (\Box_l^{\downarrow z}(A\backslash_{ca}B))\$ \rightarrow (\Box_l^{\downarrow x}A\backslash_{ca}\Box_l^{\downarrow y}B)\$$$

avec  $x, y, z$  tels que  $x \preceq y \prec z$

Noter la place différente des valeurs de paramètre  $x, y, z$ . Cependant, nous n'aurons pas besoin de cette règle dans le cadre de la transposition de l'analyse de Kraak, aucune des catégories intervenant ne portant un slash  $\backslash_{ca}$ .

La règle structurelle  $K_u$ , très similaire à  $\mathbf{K}$ , engendre sur le même principe la règle suivante (nous supprimons les  $\$$  pour plus de lisibilité, mais cette généralisation reste valide) :

<sup>25</sup>. Bien que les situations présentées ci-après ne le nécessitent pas, il est facile de se rendre compte que cette opération serait également nécessaire pour les règles  $\mathbf{B}$ .

$$K_u \text{ GCC-MM} : \Box_l^{\downarrow 8}(A|_{ca}B) \rightarrow \Box_l^{\downarrow 8}A|_{ca}\Box_l^{\downarrow 8}B$$

Avec  $| \in \{/, \backslash\}$  (comme auparavant, le cas de  $\backslash$  est envisagé à la suite d'une volonté de généraliser les règles, mais en réalité aucune catégorie du lexique utilisé en 2.3 ne comporte de slash  $\backslash_u$ ).

Pour l'analyse incrémentale, comme pour  $K$ , nous avons besoin d'introduire une règle  $D$  intégrant l'effet de  $K_u$ , qui ne diffère de la précédente que par la condition sur  $y, z$  :

$$\triangleright D_{K_u} : X|_{ca}\Box_l^{\downarrow 8}(Y|_cZ) \Box_l^{\downarrow 8}(Y|_uW) \rightarrow X|_u\Box_l^{\downarrow 8}(W|_cZ)$$

Le reste des règles se dérive aussi facilement. Le cas des règles  $K1$  et  $K2$  est intéressant, car une seule des structures constituant le produit de la prémisse est tenue de porter la modalité  $p$ , ainsi il se peut que la catégorie opérateur prémisse ne porte aucune modalité. Voir par exemple la preuve ci-dessous :

$$\frac{\frac{\Delta \vdash X/Y \quad \frac{u \vdash \Box_p^{\downarrow x}Y}{\langle u \rangle_p^x \vdash Y} \Box_p^{\downarrow x}E}{\Delta \circ \langle u \rangle_p^x \vdash X} /E}{\frac{\langle \Delta \circ u \rangle_p^x \vdash X}{\Delta \circ u \vdash \Box_p^{\downarrow x}X} K2}{\Delta \circ u \vdash \Box_p^{\downarrow x}X} \Box_p^{\downarrow x}I}{\Delta \vdash \Box_p^{\downarrow x}X /I} /I$$

On voit donc qu'on peut dériver  $\Box_p^{\downarrow x}X / \Box_p^{\downarrow x}Y$  à partir de la catégorie sans modalité  $X/Y$ . Par trois démonstrations similaires, on obtient les règles  $\text{GCC-MM}$  ci dessous :

$$\begin{aligned} K1 \text{ GCC-MM} &: \Box_p^{\downarrow x}(A|_cB) \rightarrow \Box_p^{\downarrow x}A|_cB \\ K2 \text{ GCC-MM} &: A|_cB \rightarrow \Box_p^{\downarrow x}A|_c\Box_p^{\downarrow x}B \end{aligned}$$

Les preuves des règles suivantes à partir des règles structurelles  $I, \emptyset$  sont triviales, nous nous contenterons d'énoncer les règles  $\text{GCC-MM}$  :

$$\begin{aligned} I \text{ GCC-MM} &: \Box_l^{\downarrow x}X \rightarrow \Box_p^{\downarrow x}X \\ \emptyset \text{ GCC-MM} &: \Box_j^{\downarrow 8}X \rightarrow \Box_j^{\downarrow 0}X \\ &\text{avec } j \in \{l, p\} \end{aligned}$$

Toutes ces règles peuvent être généralisées de la manière vue précédemment, en ajoutant un  $\$$  après chaque prémisse et la conclusion, comme le lecteur pourra le vérifier facilement en reprenant le principe des preuves précédentes.

Il faut en revanche signaler une différence notable concernant la règle  $\mathbf{MA}$  vue dans la partie 2.3 (règle d'associativité).  $\text{CTL}$  suppose généralement un régime d'associativité plus strict que  $\text{GCC-MM}$ . L'associativité est par défaut désactivée en  $\text{CTL}$ , il est nécessaire de préciser les modalités associatives en leur attribuant les règles structurelles adéquates. En  $\text{GCC-MM}$ , à l'inverse, les règles de composition harmoniques (qui nécessitent la propriété d'associativité) sont souvent considérées comme universellement valides (sauf en cas d'utilisation de la modalité  $\star$  de [2]). Soucieux de pouvoir maintenir d'autre part l'incrémentalité de l'analyse (qui nécessite un haut degré d'associativité), nous utiliserons le « degré d'associativité » habituel de  $\text{GCC}$  induit par les règles  $\mathbf{B}$ , étendu par les règles  $\mathbf{D}$  vues dans la partie précédente (ce que nous traduirons en  $\text{CTL}$  en liant toutes les modalités aux règles structurelles d'associativité). Ceci nous expose possiblement à la survenance de surgénérations du type 3.4, dans des développements ultérieurs, il importerait de bien évaluer précisément les causes de ces surgénérations par associativité, en voyant s'il est possible de les éviter autrement qu'en restreignant l'associativité, ou tout du moins en s'assurant que les éventuelles restrictions de l'associativité ne restreindraient pas l'incrémentalité de l'analyse.

Nous laissons volontairement de côté la modalité  $\diamond$  utilisée pour l'analyse des verbes interdisant la montée des cliques (comme *vouloir*). En effet cette méthode (qui impose à un verbe de consommer son objet avant d'être consommé par son auxiliaire) va à l'encontre du but que nous nous sommes fixés, à savoir de maintenir une analyse incrémentale. Des investigations pourraient très utilement explorer les éventuels moyens de rétablir

la couverture que permettait l'introduction de  $\diamond$  (le type des verbes interdisant la montée des clitiques) tout en maintenant la possibilité d'une analyse incrémentale.

Nous avons donc, à partir des règles structurelles utilisées par [15], dérivé un ensemble de règles GCC-MM capturant une partie des effets induits par ces règles structurelles. Le pari effectué est que les effets capturés seront suffisants pour atteindre la couverture souhaitée des phénomènes linguistiques modélisés (si cela échoue, il sera nécessaire d'ajouter de nouvelles règles, ou de généraliser des règles existantes, en s'efforçant de garder l'ensemble des règles le plus petit possible). L'avantage obtenu étant que, puisqu'à tout calcul GCC-MM peut être associée une preuve CTL, toute surgénération survenant dans GCC-MM surviendra *aussi* dans CTL, donc si on élimine la surgénération de la modélisation CTL (ce qui est plus facile du fait de la meilleure expressivité du système), elle sera également éradiquée du système GCC-MM.

Il ne nous reste plus qu'à transposer l'opérateur d'extraction  $\uparrow$ , qui n'est pas directement définissable dans GCC-MM. Cette discussion a déjà été effectuée dans la partie précédente, et nous avons justifié que, si des mécanismes plus fins sont sans doute possibles (mais nécessitent une certaine réflexion), une solution temporaire acceptable peut être de recourir à l'ambiguïté lexicale, en énumérant l'action des catégories portant un opérateur  $\uparrow$  sur leurs différents arguments possibles.

Voici les types que nous avons obtenus après énumération :

- les clitiques complément d'objet direct (*le, la, les, en...*) se verront attribuer les types  $SV/ca(SV/cNP_{acc})$  et  $(SV/cNP_{dat})/ca(SV/cNP_{dat}/cNP_{acc})$
- les autres clitiques se verront attribuer le type simple  $SV/ca(SV/cNP)$  (assorti des habituels traits de cas, temps...) pour les clitiques qui extraient un objet, ou  $SV/caSV$  pour *ne, y*.

On a vu dans le paragraphe précédent que ces types ajoutés conservent également la propriété de non-surgénération, puisque les assignations supplémentaires peuvent elles aussi être vues comme des raccourcis de preuves CTL.

Le calcul des structures prédicatives n'est pas l'objet principal du présent rapport, nous nous bornerons à suivre [23] pour l'attribution des interprétations des clitiques objets. Un clitique objet est vu comme un opérateur qui prend en argument un prédicat non saturé (un verbe auquel il manque un objet) et le sature en lui faisant consommer une variable libre (qui apparaît donc à la place du groupe nominal que le clitique objet remplace). Par exemple *la* se verra attribuer (en logique combinatoire) le type :  $(C_* x_f)$  (où  $x_f$  -f pour féminin-est la variable libre introduite par le clitique).

## 6.2 Résultats

Nous reprenons maintenant les analyses CTL effectuées dans 2.3, en utilisant notre système GCC-MM construit à l'étape précédente. Nous analysons également quelques exemples simples supplémentaires.

Analyse de la phrase : *Miguelito le regarde*. (comme version contractée de *Miguelito regarde le ciel* par exemple...). Comme auparavant, on posera  $SV=_{def}S/cNP$ . L'indice  $GCC-MM$  sera omis dans la notation des règles. Noter que nousinstancions le paramètre de la modalité  $l$  (et  $p$ ) directement aux valeurs adéquates. En pratique, l'analyseur devra être capable de retarder son instanciation en étudiant les différentes contraintes qui pèsent sur lui (qui ne sont que des systèmes d'inéquations basiques découlant de  $K$ ). L'emploi d'une stratégie d'héritage peut probablement se révéler efficace pour réaliser cette tâche (les catégories devenant de plus en plus spécifiées à mesure que s'ajoutent des contraintes).

$$\begin{array}{c}
\frac{\text{Miguelito}}{\text{NP : Miguelito}} \\
\frac{\text{S/cSV : } C_* \text{ Miguelito}}{\square_p^{j0}S/c\square_p^{j0}SV : C_* \text{ Miguelito}} >T \\
\frac{\square_p^{j0}S/c\square_p^{j0}SV : C_* \text{ Miguelito}}{\square_p^{j0}S/ca\square_l^{j8}(SV/cNP_{acc}) : B (C_* \text{ Miguelito}) (C_* x_m)} >K2
\end{array}
\quad
\begin{array}{c}
\frac{\text{le}}{\square_l^{j4}(SV/ca(SV/cNP_{acc})) : C_* x_m} \\
\frac{\square_l^{j4}(SV/ca(SV/cNP_{acc})) : C_* x_m}{\square_l^{j0}SV/ca\square_l^{j8}(SV/cNP_{acc}) : C_* x_m} K \\
\frac{\square_l^{j0}SV/ca\square_l^{j8}(SV/cNP_{acc}) : C_* x_m}{\square_p^{j0}SV/ca\square_l^{j8}(SV/cNP_{acc}) : C_* x_m} I \\
\frac{\square_p^{j0}SV/ca\square_l^{j8}(SV/cNP_{acc}) : C_* x_m}{\square_p^{j0}S : B (C_* \text{ Miguelito}) (C_* x_m) \text{ regarde}} >B
\end{array}
\quad
\frac{\text{regarde}}{\square_l^{j8}(SV/NP_{acc}) : \text{regarde}} >B$$

Remarquons que la règle  $K$  est correctement appliquée puisque  $0 \preceq 4 \prec 8$ . On obtient bien la catégorie désirée  $\square_p^{j0}S$  à l'issue du calcul : la phrase est correctement validée comme étant syntaxiquement bien formée.



On voit que l'intervention de la règle  $K_u$  oblige le constituant  $J'ai$  à prendre en argument un  $SV_{pp}$  dont le paramètre de la modalité est 8, ce qui est rendu impossible par l'intervention du clitique  $la$  entre l'auxiliaire et le verbe (qui fait baisser le paramètre à 4).

$$\begin{array}{c}
\frac{\frac{\text{Je}}{\text{NP}}}{\text{S/cSV}} > \mathbf{T} \quad \frac{\frac{\text{l'}}{\frac{\frac{\square_l^{\downarrow 4}(\text{SV/ca}(\text{SV/cNP}_{acc}))}{\square_l^{\downarrow 0}\text{SV/ca}\square_l^{\downarrow 8}(\text{SV/cNP}_{acc})} \text{K}}{\square_p^{\downarrow 0}\text{SV/ca}\square_l^{\downarrow 8}(\text{SV/cNP}_{acc})} \text{I}}{\square_p^{\downarrow 0}\text{S/c}\square_p^{\downarrow 0}\text{SV}} \text{K2}}{\square_p^{\downarrow 0}\text{S/ca}\square_l^{\downarrow 8}(\text{SV/cNP}_{acc})} > \mathbf{B} \quad \frac{\text{ai}}{\square_l^{\downarrow 8}(\text{SV/uSV}_{pp})} > \mathbf{D}_{K_u} \quad \frac{\text{trouvée}}{\square_l^{\downarrow 8}(\text{SV}_{pp}/\text{cNP}_{acc})} > \\
\frac{\square_p^{\downarrow 0}\text{S/u}\square_l^{\downarrow 8}(\text{SV}_{pp}/\text{cNP}_{acc})}{\square_p^{\downarrow 0}\text{S}} >
\end{array}$$

Notons que la montée du clitique n'altère en rien le bon déroulement du calcul de la structure prédicative. En reprenant les étapes pas à pas, on voit qu'on a effectué :

- >  $\mathbf{T}$  sur  $Je$  :  $\mathbf{C}_* \mathbf{Je}$
- >  $\mathbf{B}$  entre  $Je$  et  $l'$  :  $\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)$
- >  $\mathbf{D}_{K_u}$  entre  $Je l'$  et  $ai$  :  $\mathbf{D}(\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)) \text{ ai}$
- > entre  $Je l'ai$  et  $trouvée$  :  $\mathbf{D}(\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)) \text{ ai trouvée}$

Réduisons cette dernière expression :

$$\begin{aligned}
\mathbf{D}(\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)) \text{ ai trouvée} &= \mathbf{C} \mathbf{B}^2 \mathbf{B}(\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)) \text{ ai trouvée} \\
&= \mathbf{B}^2(\mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)) \mathbf{B} \text{ ai trouvée} \\
&= \mathbf{B}(\mathbf{C}_* \mathbf{Je})(\mathbf{C}_* x_f)(\mathbf{B} \text{ ai trouvée}) \\
&= \mathbf{C}_* \mathbf{Je}((\mathbf{C}_* x_f)(\mathbf{B} \text{ ai trouvée})) \\
&= \mathbf{C}_* x_f(\mathbf{B} \text{ ai trouvée}) \mathbf{Je} \\
&= \mathbf{B} \text{ ai trouvée } x_f \mathbf{Je} \\
&= \mathbf{ai}(\text{trouvée } x_f) \mathbf{Je}
\end{aligned}$$

Terminons en donnant l'analyse d'une phrase un peu plus complexe : *Manolito la lui fait goûter* (condensé de *Manolito fait goûter une olive à Mafalda* par exemple).

$$\begin{array}{c}
\frac{\frac{\text{Manolito}}{\text{NP}}}{\text{S/cSV}} > \mathbf{T} \quad \frac{\frac{\text{la}}{\frac{\frac{\square_l^{\downarrow 4}(\text{SV/ca}(\text{SV/cNP}_{acc}))}{\square_l^{\downarrow 0}\text{SV/ca}\square_l^{\downarrow 5}(\text{SV/cNP}_{acc})} \text{K}}{\square_p^{\downarrow 0}\text{SV/ca}\square_l^{\downarrow 5}(\text{SV/cNP}_{acc})} \text{I}}{\square_p^{\downarrow 0}\text{S/c}\square_p^{\downarrow 0}\text{SV}} \text{K2}}{\square_p^{\downarrow 0}\text{S/ca}\square_l^{\downarrow 5}(\text{SV/cNP}_{acc})} > \mathbf{B} \quad \frac{\text{lui}}{\square_l^{\downarrow 5}(\text{SV/ca}(\text{SV/cNP}_{dat}))} > \mathbf{D}_K \\
\frac{\square_p^{\downarrow 0}\text{S/u}\square_l^{\downarrow 8}((\text{SV/cNP}_{dat})/\text{cNP}_{acc})}{\square_p^{\downarrow 0}\text{S}} >
\end{array}$$

$$\begin{array}{c}
\frac{\frac{\text{Manolito la lui}}{\square_p^{\downarrow 0}\text{S/u}\square_l^{\downarrow 8}((\text{SV/cNP}_{dat})/\text{cNP}_{acc})} \quad \frac{\text{fait}}{\square_l^{\downarrow 8}(\text{SV/cNP}_{dat}/\text{uSV}_i)}}{\square_p^{\downarrow 0}\text{S/u}\square_l^{\downarrow 8}(\text{SV}_i/\text{cNP}_{acc})} > \mathbf{D}_{K_u} \quad \frac{\text{goûter}}{\square_l^{\downarrow 8}(\text{SV}_i/\text{cNP}_{acc})} > \\
\frac{\square_p^{\downarrow 0}\text{S}}{\square_p^{\downarrow 0}\text{S}} >
\end{array}$$

Une fois encore, la construction syntaxique de la phrase est bien validée. Tout ordre alternatif incorrect (*\*Manolito lui la fait goûter*, *\*Manolito lui fait la goûter*, *\*Manolito fait la lui goûter...*), serait bloqué car ils l'étaient dans l'analyse CTL et que toutes les règles utilisées ici sont des théorèmes CTL.

En reprenant les étapes du calcul, on pourra voir qu'il construit la structure prédicative :

## **D (D (B (C\* Manolito) (C\* $x_{f,acc}$ )) (C\* $x_{dat}$ )) fait goûter**

Dont le lecteur pourra vérifier qu'elle se réduit en :

**fait (goûter  $x_{f,acc}$ )  $x_{dat}$  Manolito**

Une telle expression pourrait ensuite être utilisée à un niveau supérieur d'analyse pour déterminer à quelles entités se rapportent les variables  $x_{f,acc}$ ,  $x_{dat}$  compte tenu du contexte de la phrase.

### **6.3 Bilan et travaux futurs**

À l'issue de cette partie, nous sommes donc parvenus à rendre compte de la plupart des phénomènes ayant trait aux clitics du français modélisés par l'analyse de Kraak au sein d'un simple système GCC-MM. Pour ce faire, nous avons transposé un petit nombre de théorèmes du système CTL dans GCC-MM (modulo la généralisation nécessaire du système de traits), où ils ont été stipulés comme règles combinatoires. Cette démarche s'est révélée fructueuse sur un certain nombre de points résumés ci-dessous :

- Les règles utilisées étant toutes basées sur des théorèmes de CTL, toute surgénération survenant dans le système GCC-MM est une surgénération qui existe dans le système CTL de départ. Ainsi, si le système CTL n'entraîne pas de surgénération (qualité bien plus facile à atteindre dans CTL du fait de l'expressivité du système), le système GCC-MM transposé ne sur-générera pas non plus.
- Nous sommes parvenus à maintenir une analyse incrémentale sur les exemples traités, notamment grâce à l'utilisation des règles **D**. La base CTL nous a permis de calculer leur effet sur les modalités utilisées de façon à conserver la cohérence du système (notamment pour les règles  $D_K, D_{K_u}$  introduites dans cette partie).
- L'interface syntaxe sémantique est elle aussi maintenue pour le système GCC-MM, sans difficulté particulière.
- Puisque le système construit appartient à GCC-MM, il hérite des propriétés algorithmiques de celui-ci (analysable en temps polynomial, ce qui est un atout majeur par rapport au système CTL de départ. On notera en outre que le système GCC-MM produit des dérivations plus concises que les preuves CTL. Ceci est dû au fait que les règles « compilent » plusieurs étapes de déduction CTL en une seule.

Cependant, du fait des différences entre les deux systèmes, la transposition de CTL vers GCC-MM ne peut être parfaite. Ce qui n'est pas sans soulever un certain nombre de questions et difficultés par rapport à la méthode proposée :

- Si le problème de surgénération est maîtrisé (car réductible au problème original dans CTL), la limite principale de la transposition est l'incomplétude constatée du système GCC-MM transposé par rapport au système CTL original. En d'autres termes, il n'est pas facile de prévoir dans quelles situations le système GCC-MM sous-générera par rapport au système CTL. Aussi, le choix des théorèmes à transformer est une étape tout à fait cruciale. Ceux-ci doivent être suffisamment généraux pour limiter ce problème d'incomplétude, cependant nous ne disposons pour l'instant d'aucune méthode objective pour choisir les théorèmes à retenir.
- La crainte de la duplication de règles qui a pu naître en constatant les deux versions  $\langle \mathbf{B}_\times\text{-I} \rangle, \langle \mathbf{B}_\times\text{-II} \rangle$  de la règle croisée arrière que doit introduire Steedman pour arriver à ses fins (voir [27]) se trouve confirmée voire renforcée dans le système multi-modal et le principe de transposition de CTL vers GCC-MM. On constate que l'introduction de modalités, en particulier des modalités unaires, est susceptible d'entraîner une duplication de règles, la transposition d'une règle structurelle CTL (). Témoins les règles  $D_K$  et  $D_{K_u}$

que nous avons été contraints de stipuler. Ceci découle la nécessité de traduire leur effet sur les modalités intervenant dans leurs prémisses, que l'unification ne modélise qu'inadéquatement.

Deux solutions sont possibles. La première, idéale, serait de trouver des théorèmes CTL qui « respectent » les règles GCC, le type du résultat d'une règle GCC obtenu par les mécanismes d'unification doit être équivalent au type déduit de la preuve CTL associée à la règle (or bien souvent le type GCC obtenu est moins général que le type CTL). Mais rien n'assure qu'il est possible de trouver de tels théorèmes. La deuxième solution est de se résigner à décliner l'ensemble des règles en plusieurs versions pour certaines modalités. Ce recours est bien sûr largement insatisfaisant, car il menace mal l'idéal d'universalité (et de concision) associé à l'ensemble des règles combinatoires de tels systèmes. Plusieurs questions se posent alors. L'ensemble des modalités doit-il être universel, ou chaque langue requière-t-elle un système de modalités qui lui est propre ? Quel statut faut-il donner aux versions modales multiples des règles de base ? Si l'ensemble des modalités est universel, les versions modales multiples des règles combinatoires peuvent-elles l'être aussi ?

- Certaines des règles dérivées du système CTL (notamment la règle  $K2_{GCC-MM}$  qui peut potentiellement être invoquée sur n'importe quelle catégorie fonctionnelle) nécessite l'utilisation de déclencheurs (ex. des métarègles à la façon de [7]), ou alors doivent être réévaluées comme des règles lexicales (i.e. ne pouvant être utilisées que sur les éléments du lexique en temps que « générateurs » de tous les types associés à une entrée, et pas entre des entités complexes au milieu d'une dérivation).

D'une manière générale, on aura noté que cette méthode aggrave la redondance au sein du lexique GCC-MM. Redondance au niveau des types tout d'abord, on a vu par exemple que la transposition de l'opérateur d'extraction  $\uparrow$ , ou encore le paramètre de la modalité  $\langle \rangle_l$  se traduiraient par une multiplication de types, quoique la majeure partie de l'information contenue dans ces types soit commune. La même situation est constatée au niveau des règles, de multiples versions modales sont déclinées pour spécifier de petites nuances dans le traitement des modalités, mais ces règles demeurent essentiellement le reflet d'une règle combinatoire de base.

Pour supprimer ces redondances, une piste sans doute prometteuse (et assez naturelle) serait d'envisager l'exploitation plus systématique de relations d'héritage et de propagation de traits entre les catégories du lexique et les modalités. L'utilisation du paramètre de la modalité  $\langle \rangle_l$ , par exemple induirait l'utilisation de catégories sous-spécifiées (correspondant par exemple à l'ensemble des valeurs possibles pour ledit paramètre) qui deviendraient de plus en plus déterminées au fil des contraintes apparues dans les différentes combinaisons (les valeurs possibles du paramètre se restreignant). Il pourra pour ce faire être judicieux de s'inspirer de techniques utilisées dans d'autres formalismes (tel HSPG, [6]) à ce sujet.

Concernant le problème de l'universalité de l'ensemble des modalités et des versions modales des règles combinatoires, il importe d'effectuer un travail systématique de comparaison entre différentes langues, au moins d'une même famille, pour voir si un système modal défini dans l'une est réutilisable tel quel dans les autres. En d'autres termes, il s'agit de déterminer s'il est effectivement possible de faire porter par le lexique catégoriel seul toute la variabilité entre ces langues. Dans notre cas, il serait intéressant, par exemple, de tester l'adaptabilité du système proposé dans la partie 6 à l'analyse des pronoms clitiques d'autres langues romanes (c'est-à-dire de l'évaluer par rapport au quatrième critère que nous avons défini dans notre introduction).

## Conclusion

Tout au long de ce rapport, nous nous sommes efforcés de donner au lecteur un aperçu des positions relatives de GCC, GCC-MM et CTL par rapport au triple dilemme de sous-génération des grammaires, du problème inverse de surgénération, et de la complexité algorithmique associée à leur analyse. Nous avons vu comment GCC-MM, la Grammaire Catégorielle Multi-Modale de J. Baldrige [2], tente de jeter un pont entre le point de vue des systèmes à base de règles, tels GCC, et celui des systèmes déductifs comme CTL, par l'introduction de la notion de modalité dans GCC, et l'interprétation de ses règles combinatoires comme des preuves CTL « compilées ».

Quoiqu'il constitue à lui seul un apport non négligeable en matière de souplesse et d'universalité des gram-

maires (notamment au regard des anciens mécanismes de restriction de règles), on aura pu se convaincre que le système particulier de modalités proposé par [2] est certainement perfectible. En témoignent les lacunes mises en évidence par l'auteur lui-même, reprises et complétées dans ce rapport (cas de surgénérations/sous-générations intriquées, utilité discutable de la notion de directionnalité...).

En définitive, c'est surtout la méthode par laquelle le système a été obtenu que l'on retiendra. La construction des modalités de MM-GCC et l'introduction des règles **D** ont montré les services que peut rendre un fondement logique CTL, utilisé comme « microscope » pour analyser de façon minutieuse les propriétés d'une règle (MM-)GCC, et la faire interagir de façon cohérente avec un certain système de modalités.

Dans les deux dernières parties de ce rapport, nous avons entrepris d'approfondir cette correspondance entre les deux formalismes pour l'envisager dans un cadre plus systématique. Nous avons montré comment l'idée de transposition de CTL vers GCC pouvait être plus généralement exploitée pour décorréliser explicitement la lutte contre la surgénération dans GCC-MM de la lutte contre la sous-génération, et ainsi sortir du jeu inextricable de leurs rétro-actions. Concrètement, lorsque l'on cherche à modéliser un certain ensemble de phénomènes linguistiques à l'aide d'une grammaire catégorielle multi-modale, les différentes étapes associées à cette méthode sont les suivantes :

1. Modéliser les phénomènes à traiter au sein du système CTL. Cette étape est de loin plus facile que dans GCC-MM grâce à la meilleure expressivité du système, la puissance de sa base logique, et la panoplie de moyens de contrôle qu'il offre (règles structurelles modales).
2. Lorsque la modélisation est jugée satisfaisante, transposer l'analyse CTL obtenue dans le système GCC-MM. Pour ce faire, chercher des théorèmes CTL à intégrer dans GCC-MM de manière à couvrir la plus grande partie possible de l'analyse originale. L'intégration de ces théorèmes peut mener à la définition de **nouvelles règles modales** GCC-MM, ou à l'exploitation de l'**ambiguïté lexicale**, i.e. l'attribution de types multiples à certaines entrées du lexique pour couvrir les cas engendrés par l'analyse CTL. Une troisième non explorée dans ce rapport pourrait être de se servir de théorèmes CTL comme fondement à la définition de structures d'héritage entre les types de GCC-MM.

Puisque les règles GCC-MM utilisées seront aussi des théorèmes du système CTL de départ, la lutte contre la surgénération dans le système GCC-MM est en fait effectuée à l'étape 1), tandis que celle contre la sous-génération a bien lieu principalement au cours de l'étape 2), par le choix des théorèmes à intégrer dans MM-GCC. Nous pouvons ainsi chercher à augmenter la couverture de la grammaire par l'introduction de nouveaux types ou de nouvelles règles de façon contrôlée, libérés de la crainte d'introduire involontairement de nouvelles surgénérations. De par cette transposition, on peut tirer parti (indirectement) de la puissance d'expression du formalisme CTL, tout en profitant des propriétés algorithmiques attractives de GCC-MM.

Bien sûr, cette méthode ici idéalisée peut se heurter à un certain nombre d'obstacles découlant des différences entre les deux systèmes lors de sa mise en pratique. Il est à noter tout d'abord que nous ne disposons d'aucun moyen sûr de choisir les « bons » théorèmes CTL à introduire dans le système GCC-MM comme règles modales ou types supplémentaires. En outre, ces choix de théorèmes présentent aussi des phénomènes de rétro-action difficilement calculable. On a vu par exemple dans la partie 6 que l'utilisation des règles **K** et  $K_u$  impliquait l'introduction de nouvelles versions de la règle **D** pour atteindre une couverture satisfaisante. Un théorème CTL (c'est-à-dire, une règle GCC-MM) ne se choisit donc pas uniquement pour lui-même, mais en fonction d'un certain ensemble de règles déjà intégrées au système.

Un autre type d'obstacle courant dans le cas plus spécifique d'une transposition en GCC-MM d'une analyse CTL existante, est une incompatibilité des buts visés. Typiquement, les analyses CTL se soucient généralement peu d'établir une analyse incrémentale (la pertinence de cette notion est du reste limitée en CTL du fait du mécanisme de preuve et de l'introduction d'hypothèses), et donc n'hésite pas à modéliser certains phénomènes en introduisant une restriction de l'associativité (par ex. ici, les verbes interdisant la montée des clitiques), ce que ne peut se permettre un système MM-GCC désireux de conserver l'incrémentalité de l'analyse. À noter que ce genre d'incompatibilité n'est pas spécifique à CTL, puisque J. Baldrige lui-même introduit une modalité non-associative dans son système. À ce sujet, il serait important de mener une étude détaillée des causes des surgénérations engendrées par l'associativité et de la question de sa restriction, dans la perspective de la conservation d'une analyse incrémentale.



Malgré ces difficultés découlant des différences irréductibles existantes entre les propriétés formelles des deux systèmes et les buts concrets auxquels ils sont généralement associés, on a vu tout au long du rapport qu'il était possible de dégager de fortes analogies entre les mécanismes utilisés d'un côté et de l'autre, et de les exploiter pour tenter de combiner leurs avantages respectifs. On peut cependant s'interroger sur le *sens* d'une telle utilisation conjointe. Si on choisit de modéliser les phénomènes de la langue d'abord via CTL, comment interpréter l'incomplétude de GCC-MM par rapport au formalisme original, et donc à ces phénomènes ? Si GCC-MM n'est pas capable de capturer certains concepts linguistiques dans toute leur généralité (exemple, l'opérateur d'extraction  $\uparrow$  utilisé par Kraak), pourquoi conserver ce formalisme ? Il y a certes l'argument de la complexité algorithmique, mais ne serait-il pas plus naturel de chercher directement à améliorer les algorithmes d'analyse de CTL plutôt que d'utiliser une telle transposition imparfaite ?

Une réponse possible est de réviser notre position vis-à-vis de l'impératif de « complétude ». De la même façon qu'on peut savoir que Socrate était mortel sans être à proprement parler logicien, il nous est possible de *pratiquer* une langue sans être linguiste, sans forcément les comprendre les « systèmes » mis en jeu dans toute leur généralité. Il n'y a donc pas forcément d'incohérence à voir des analyses « complètes », atteignant un haut degré d'abstraction, tel un système CTL, s'instancier avec pertes dans des mécanismes plus simples, par exemple un ensemble de *règles* MM-GCC. Mécanismes qui ne permettent peut-être pas d'abstraire les phénomènes impliqués dans toute leur généralité, mais qui constituent néanmoins une base fonctionnelle permettant de « pratiquer » un sous-ensemble de la langue totale (sous-ensemble que l'on peut espérer coïncider avec celui des cas réellement utiles, par ex. les phrases contenant moins de quatre relatives emboîtées). Le paradoxe ci-dessus peut donc être résolu si l'on décide de voir en CTL un formalisme « d'introspection », qui peut servir à dégager et formaliser des généralités linguistiques, et en MM-GCC un formalisme d'« action » (ou de routine) dans lequel ces généralités sont compilées (peut-être imparfaitement) en des règles d'usage fonctionnelles.

## Références

- [1] K. Ajdukiewicz. Die syntaktische Konnexität. *Studia philosophica*, 1(1) :27, 1935.
- [2] J. Baldridge. Lexically specified derivational control in Combinatory Categorical Grammar. 2002.
- [3] J. Baldridge and G.J.M. Kruijff. Multi-modal combinatory categorial grammar. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 211–218. Association for Computational Linguistics, 2003.
- [4] Jason Baldridge. Categorical grammar. In Tibor Kiss and Artemis Alexiadou, editors, *Handbook of Syntax*. Berlin : de Gruyter, (to appear).
- [5] Y. Bar-Hillel. A quasi-arithmetical notation for syntactic description. *Language*, 29(1) :47–58, 1953.
- [6] J. Beavers. Type-inheritance combinatory categorial grammar. In *Proceedings of the 20th international conference on Computational Linguistics*, pages 57–es. Association for Computational Linguistics, 2004.
- [7] I. Biskri. *La Grammaire catégorielle combinatoire applicative dans le cadre de la grammaire applicative et cognitive*. PhD thesis, Thèse de Doctorat, EHESS, Paris, 1995.
- [8] I. Biskri and J.P. Desclés. Applicative and Combinatory Categorical Grammar (from syntax to functional semantics). *Recent Advances in Natural Language Processing (selected Papers of RANLP 95)*, pages 71–84, 1997.
- [9] I. Biskri and J.P. Desclés. Analyse de la coordination et de la subordination au moyen de la grammaire catégorielle combinatoire applicative. In *Colloque Subordination-Coordination*, 2005.
- [10] J.P. Desclés. Langages applicatifs, langues naturelles et cognition. 1990.
- [11] J.P. Desclés and I. Biskri. Logique combinatoire et linguistique : grammaire catégorielle combinatoire applicative. *Mathématiques Informatique et Sciences Humaines*, pages 39–68, 1995.
- [12] Curry B. H. Fey, R. Combinatory logic. 1958.
- [13] J.Y. Girard. Linear logic\* 1. *Theoretical computer science*, 50(1) :1–101, 1987.

- [14] F. Hoyt, J. Baldridge, et al. A logical basis for the d combinator and normal form in ccg. *Proceedings of ACL-08 : HLT*, pages 326–334, 2008.
- [15] E. Kraak. A deductive account of French object clitics. *SYntax and Semantics*, pages 271–312, 1998.
- [16] G.J.M. Kruijff, P. Lison, T. Benjamin, H. Jacobsson, and N. Hawes. Incremental, multi-level processing for comprehending situated dialogue in human-robot interaction. In *Language and Robots : Proceedings from the Symposium (LangRo ?2007)*, pages 55–64, 2007.
- [17] M. Kuhlmann, A. Koller, and G. Satta. The importance of rule restrictions in ccg. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 534–543. Association for Computational Linguistics, 2010.
- [18] J. Lambek. The mathematics of sentence structure. *American mathematical monthly*, 65(3) :154–170, 1958.
- [19] M. McConville. Incremental natural language understanding with combinatory categorial grammar.
- [20] M. McConville. Inheritance and the ccg lexicon. In *Proceedings of the 11th*, 2006.
- [21] R. Montague. The proper treatment of quantification in ordinary English. *Approaches to natural language*, 49 :221–242, 1973.
- [22] M. Moortgat. Categorial type logics. *Handbook of logic and language*, pages 93–177, 1997.
- [23] R. Moot and C. Retoré. Les indices pronominaux du français dans les grammaires catégorielles. *Linguisticae Investigationes*, 29(1) :137–146, 2006.
- [24] R.C.A. Moot. *Proof nets for linguistic analysis*. Citeseer, 2002.
- [25] G. Morrill. Type logical grammar : categorial logic of signs. *Computational Linguistics*, 23(4).
- [26] M. Schönfinkel. On the building blocks of mathematical logic. In J. van Heijenoort, editor, *From Frege to Gödel*, page 357. Harvard University Press, 1967. English Translation from the German original. Includes foreword by W.V.O. Quine.
- [27] M. Steedman. *The syntactic process*. The MIT press, 2001.
- [28] M. Steedman and J. Baldridge. Combinatory categorial grammar. *Nontransformational Syntax : A Guide to Current Models*. Blackwell, Oxford, 2009.
- [29] K. Vijay-Shanker and David J. Weir. The equivalence of four extensions of context-free grammars. *Mathematical Systems Theory*, 27 :27–511, 1994.
- [30] K. Vijay-Shanker and D.J. Weir. Parsing some constrained grammar formalisms. *Computational Linguistics*, 19(4) :591–636, 1993.
- [31] Aline Villavicencio. Building a wide-coverage combinatory categorial grammar, 1997.
- [32] Aline Villavicencio. The acquisition of a unification-based generalised categorial grammar, 2002.
- [33] K. Wittenburg. Predictive combinators : a method for efficient processing of combinatory categorial grammars. In *Proceedings of the 25th annual meeting on Association for Computational Linguistics*, pages 73–80. Association for Computational Linguistics, 1987.