

Une grammaire TAG vue comme une grammaire Sens-Texte précompilée

Marie-Hélène Candito & Sylvain Kahane

TALANA, Université Paris 7, case 7003

2, place Jussieu 75251 Paris cedex 05

Courriel : marie-helene.candito@linguist.jussieu.fr, sk@ccr.jussieu.fr

Résumé

Dans cet article¹, nous comparons deux modèles linguistiques utilisés en TAL, les grammaires d'arbres adjoints [= TAG] et la Théorie Sens-Texte [= TST]. Nous montrons que ces deux modèles présentent des similitudes notables et que les représentations les plus profondes qu'ils donnent d'une phrase — la représentation sémantique en TST et l'arbre de dérivation en TAG — sont équivalentes. De ce rapprochement découle d'une part que l'on peut utiliser la procédure de dérivation et la grammaire TAG pour opérer la correspondance Sens-Texte, et d'autre part que l'on peut concevoir une grammaire TAG comme le résultat de la précompilation d'une grammaire Sens-Texte.

Introduction

TAG est un formalisme initialement développé dans le cadre de la Théorie des langages formels, dont la pertinence pour la représentation de phénomènes linguistiques a été argumentée après coup (Kroch & Joshi 85). L'aspect procédural a été étudié en profondeur et des implémentations à large échelle ont été réalisées pour l'analyse ou la génération automatique (XTAG 95; Danlos & Meunier 96). Si la lexicalisation forte du formalisme a apporté des avantages tant linguistiques que computationnels (Abeillé 91), elle entraîne une redondance dans la représentation de la grammaire, sans qu'une organisation modulaire vienne y remédier.

A l'inverse, la TST est une théorie qui d'une part est modulaire et d'autre part a été développée par des linguistes en élaguant volontairement toutes les questions procédurales. Diverses implémentations basées sur la TST ont certes été réalisées (GOSSIP, FOG, AlethGen, ...), mais l'aspect procédural de la correspondance Sens-Texte, comme le formalisme des règles de correspondance n'ont fait l'objet que d'un début d'étude (Polguère 90; Kahane & Mel'cuk 97).

Cependant ces deux modèles ont des points de convergence notables (outre la lexicalisation forte et le traitement différencié des arguments et des modifieurs, chaque entrée lexicale en TAG et TST correspond à une unité sémantique, dont sont déclarés les actants (Abeillé 91)). Ces similitudes rendent possibles une comparaison et un enrichissement de chacun des deux modèles. La comparaison permet, pour TAG, de mieux spécifier les notions linguistiques utilisées, de réinterpréter le résultat de l'analyse TAG comme une analyse sémantique et de calquer l'organisation modulaire de la TST pour construire la grammaire. Pour la TST, cela permet de s'inspirer de la procédure de dérivation TAG pour obtenir une procédure de correspondance Sens-Texte, et de mettre en évidence les points qui doivent être développés pour que la TST devienne un modèle de référence en linguistique computationnelle, comme l'est déjà TAG.

¹ Nous remercions Anne Abeillé, Laurence Danlos, Lidija Iordanskaja, Igor Mel'cuk, Alain Polguère, Owen Rambow et nos trois referees pour leurs nombreuses remarques et suggestions faites sur la première version de cet article.

Dans les Sections 1 et 2, nous présenterons la TST, puis TAG. Nous avons intégré, dans ces présentations, les améliorations dont ont bénéficié chacun des deux modèles au cours de cette étude. Dans la Section 3, nous montrerons que les deux modèles utilisent la même notion de dépendance sémantique, bien que de manière implicite en TAG. Ceci nous permet de comparer les représentations linguistiques des deux modèles (et de montrer à quel point elles sont proches). Ce parallèle permet, d'une part, d'utiliser la procédure de dérivation TAG pour assurer la correspondance entre le niveau sémantique et la phrase en TST (Section 4) et, d'autre part, de s'inspirer de la représentation modulaire des informations linguistiques en TST pour spécifier la création d'arbres élémentaires TAG (Section 5).

1. Introduction à la TST

1.1. Les différents niveaux de représentation en TST

Dans cette étude, nous considérerons 4 des 7 niveaux de représentations de la TST: sémantique, syntaxique profond, syntaxique de surface et morphologique profond. A chaque niveau, la représentation d'une phrase est constituée d'une structure centrale et de structures périphériques qui lui sont superposées. A part pour le niveau sémantique, seule la structure centrale sera considérée.

La **représentation sémantique** [= **RSém**] d'une phrase est une représentation du sens langagier de la phrase. La structure centrale de la RSém, la **structure sémantique** [= **SSém**], est un **graphe** dont les nœuds sont étiquetés par des **sémantèmes** de la langue considérée (un sémantème étant le signifié d'une lexie) et dont les arcs sont numérotés de façon à distinguer les différents arguments d'un même sémantème prédicatif. Dorénavant, le terme **dépendance sémantique** désignera la dépendance entre sémantèmes au sens de la TST (Zolkovskij & Mel'cuk 67; Mel'cuk 88). Sur la SSém, viennent se superposer un certain nombre de marques de nœuds ou de sous-graphes constituant la **structure communicative sémantique** [= **SComm-Sém**] (thème-rhème, focalisation, arrière-plan, ...). La Fig. 1 donne la RSém pour *la nouvelle bibliothèque possède le livre que Pierre pense que Jean cherche* (notre phrase de référence). (Notation: Le mot *cherche* est une forme de la lexie CHERCHER dont le signifié est 'chercher'.)

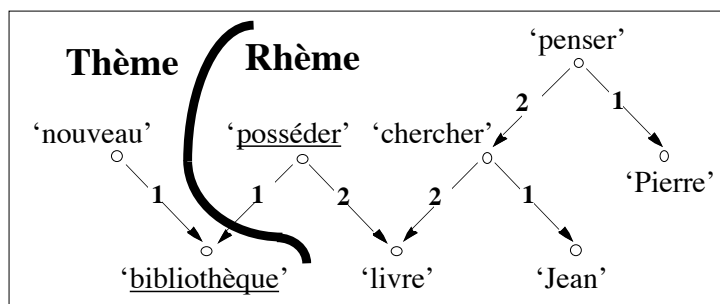


Figure 1 : la RSém pour
La nouvelle bibliothèque possède le livre que Pierre pense que Jean cherche.

La **structure syntaxique profonde** [= **SSyntP**] est un **arbre de dépendance** dont les nœuds sont étiquetés par des **lexies pleines** (accompagnées éventuellement de grammèmes) et dont les branches sont étiquetées par des relations syntaxiques universelles: six relations actancielles (I, II, ...,VI), la relation attributive ATTR, la relation de coordination COORD et la relation appenditive APPEND.

La structure **syntactique de surface** [= **SSyntS**] est un arbre de dépendance dont les nœuds sont étiquetés par des lexies (pleines ou vides) et dont les branches sont étiquetées par des relations syntaxiques (= **fonctions** ou **rôles syntaxiques**) propres à chaque langue.

La structure **morphologique profonde** [= **SMorphP**] est une chaînes de lexies assorties de marques morphologiques.

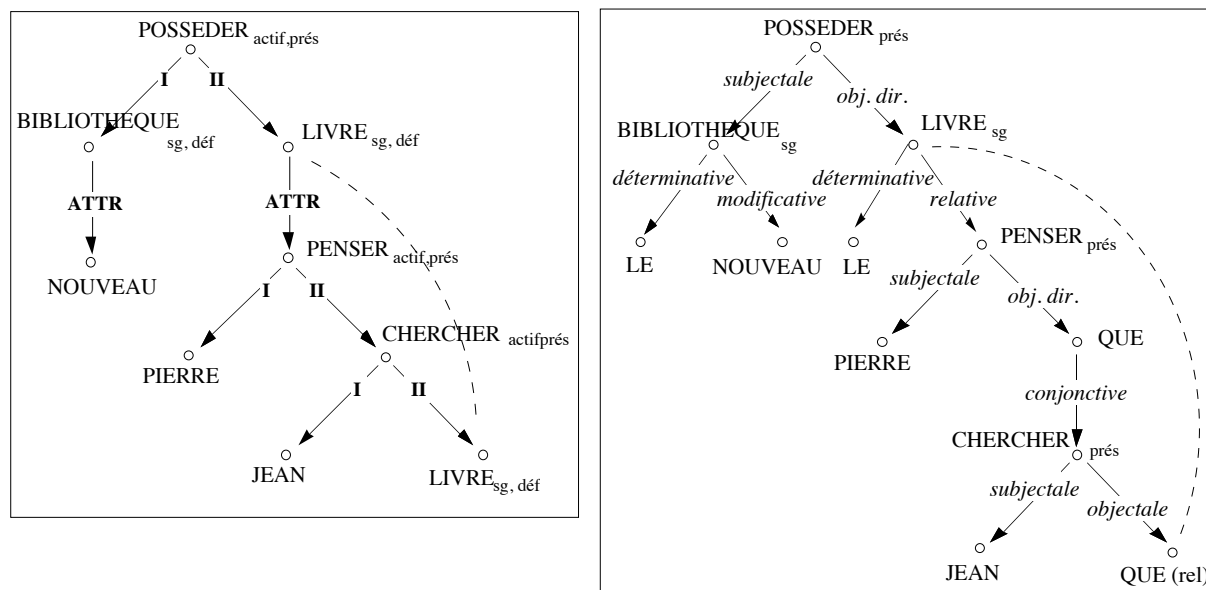


Figure 2 : SSyntP et SSyntS

1.2. Le dictionnaire et les règles de correspondance de la TST

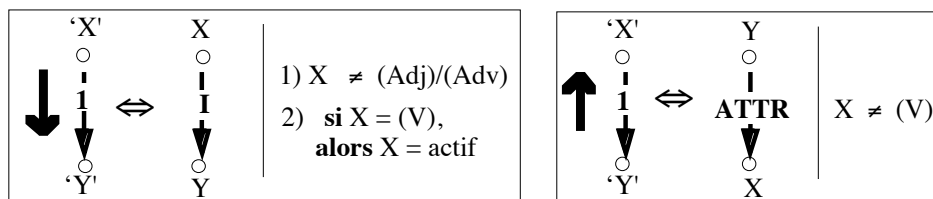
Une entrée lexicale en TST pour une lexie *L* indique le sémantème, la catégorie, la diathèse (= la correspondance entre relations sémantiques et relations syntaxiques profondes) de base et comporte un certain nombre de traits comme <non_passivable>, <verbe_pont> pour un verbe, <antéposable> pour un adjectif, etc... Par exemple pour **PENSER** on a les informations : ‘penser’, V, 1/I[N], 2/II[queV; Vinf], <non_passivable>, <verbe_pont>.

Le passage d’une RSém à une RMorphP est assuré par des règles linguistiques de correspondance (entre deux niveaux adjacents), avec consultation du dictionnaire quand nécessaire, et par des règles procédurales qui les mettent en jeu. Il nous faut considérer entre le niveau sémantique et le niveau morphologique profonds trois modules de correspondance (chacun matérialisé par une \Leftrightarrow): RSém \Leftrightarrow RSyntP \Leftrightarrow RSyntS \Leftrightarrow RMorphR. Nous présenterons les règles² de correspondance dans le sens de la synthèse, bien qu’il s’agisse de règles statiques, non orientées.

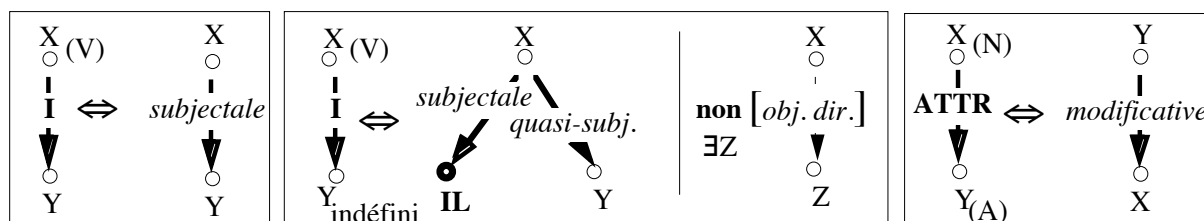
Lors de la **correspondance RSém \Leftrightarrow RSyntP** sont assurées la lexicalisation et l’arborisation. L’application des règles dépend fortement de la diathèse de la lexie (notée X) correspondant au gouverneur sémantique (noté ‘X’).³

² Le format des règles présentées suit le format adopté dans (Kahane & Mel’cuk 98); sont indiqués en gras les éléments réellement manipulés par la règle, les éléments en maigre constituant le contexte; les éléments figurant dans l’article de dictionnaire de la lexie, comme la partie du discours, sont entre parenthèses. Le signe \Leftrightarrow signifie que l’élément de gauche PEUT être exprimé par l’élément de droite (et vice versa) à condition que soient réalisées les conditions imposées par le contexte et par la partie condition placée à la droite de la règle proprement dite.

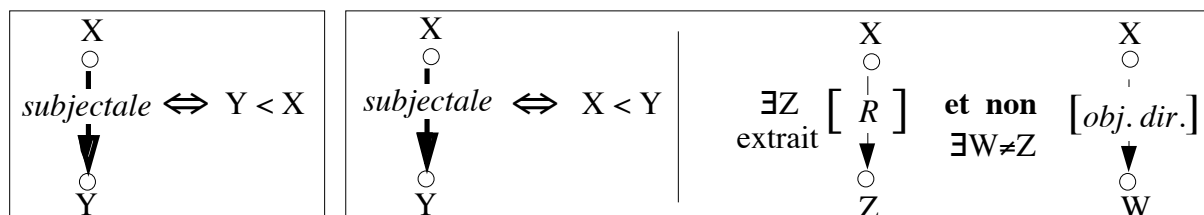
³ La grosse flèche dans la partie gauche de la règle indique le sens de parcours de l’arc qui est contraint par la SComm-Sém.

Figure 3 : Exemples de règles de correspondance RSém \Leftrightarrow RSyntP

Lors de la **correspondance SSyntP \Leftrightarrow SSyntS** sont assurées la sélection des relations syntaxiques de surface (Fig. 4), l'introduction des éventuelles prépositions régime en fonction des indications de l'entrée de dictionnaire de X et de la catégorie de Y , et la pronominalisation (grosso modo lorsqu'il y a plusieurs occurrences coréférentes en SyntP, toutes sauf une sont pronominalisées en SyntS)⁴.

Figure 4 : Exemples de règles de correspondance SSyntP \Leftrightarrow SSyntS

Lors de la **correspondance SSyntS \Leftrightarrow SMorphP** sont assurés la linéarisation — à une branche SyntS correspond grossièrement un ordre entre deux nœuds MorphP (Fig. 5) — et les accords morphologiques.

Figure 5 : Exemples de règles de correspondance SSyntS \Leftrightarrow SMorphP

2. Introduction à TAG

2.1. Les arbres élémentaires

Dans une grammaire TAG, chaque entrée lexicale L est associée à une famille d'arbres syntagmatiques, appelés **arbres élémentaires**, qui décrivent les différentes configurations possibles pour L . La lexie L **ancore** chaque arbre élémentaire de sa famille, c'est-à-dire figure comme feuille de l'arbre⁵. On distingue deux types d'arbres élémentaires: les **arbres initiaux** et les **arbres auxiliaires**. Ces deux types d'arbres se combinent avec d'autres arbres suivant des opérations différentes: les arbres initiaux par **substitution** (l'arbre vient se substituer par sa racine à une feuille d'un autre arbre) et les arbres auxiliaires par **adjonction** (l'arbre vient s'insérer à la place d'un nœud non-feuille d'un autre arbre).

⁴ De plus, l'un des nœuds d'une relative coréférent à l'antécédent de la relative doit être pronominalisé par un pronom relatif. Nous ne présentons pas les règles, d'autant que la pronominalisation n'a pas encore fait l'objet d'une réelle étude dans le cadre de la TST (et n'est pas directement abordée en TAG).

⁵ Les prépositions et conjonctions imposées par le régime de L figureront également comme feuilles des arbres élémentaires ancrés par L . D'autre part, les formes composant une expression idiomatique sont les ancres multiples d'un même arbre élémentaire.

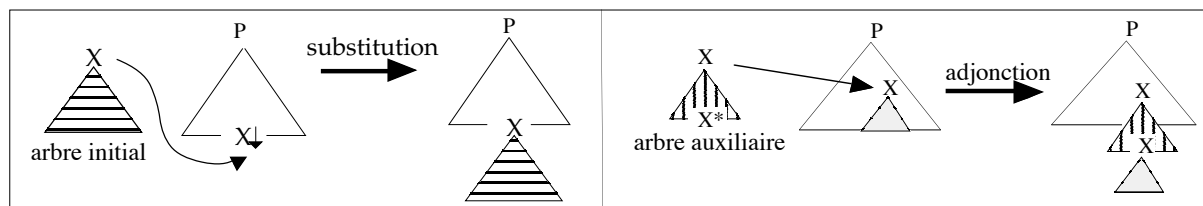


Figure 6 : la substitution et l'adjonction

Les feuilles d'un arbre initial autres que celle(s) occupée(s) par une lexie sont **des nœuds à substituer** (notés $X\downarrow$), c'est-à-dire des nœuds qui devront obligatoirement être étendus par substitution. Un arbre auxiliaire possède en plus parmi ses feuilles un **nœud dit pied** (noté X^*), de même catégorie que sa racine; c'est à ce nœud que doit être rattaché lors de l'adjonction le sous-arbre dominé par le nœud de l'arbre site d'ajonction (en tramé sur la Fig. 6).

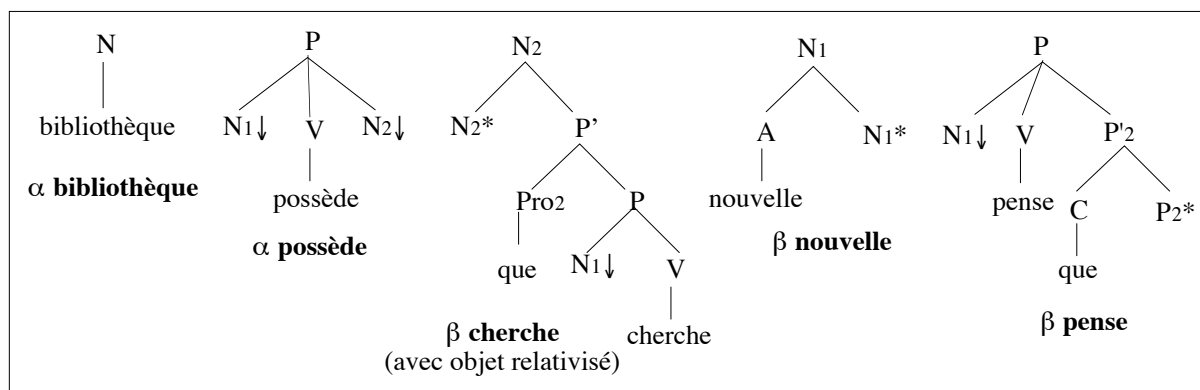


Figure 7 : Exemples d'arbres élémentaires

On peut imposer aux arbres élémentaires de respecter certains principes linguistiques (Kroch & Joshi 85; Abeillé 91; Franck 92): **lexicalisation et cohérence sémantique** (l'ensemble des éléments lexicaux à la frontière d'un arbre élémentaire correspond exactement à une unité sémantique) et **cooccurrence prédicat-arguments** (les feuilles non lexicales d'un arbre élémentaire correspondent bijectivement aux arguments sémantiques de l'ancre). Dans les exemples de la Fig. 7 (d'après (Abeillé 91)), les différentes projections des arguments de l'ancre sont numérotés comme les actants sémantiques en TST.

On notera qu'il y a deux types d'arbres auxiliaires: les **arbres auxiliaires modifieurs** dont la racine est la projection maximale du nœud pied (voir β nouvelle et β cherche) et les **arbres auxiliaires prédicatifs** dont la racine est la projection maximale de l'ancre (β pense). En termes de dépendance, cela signifie que l'ancre d'un arbre modifieur dépend syntaxiquement de l'ancre de l'arbre auquel il s'adjoint, tandis que l'ancre d'un arbre prédicatif en est le gouverneur syntaxique. Les arbres prédicatifs, qui servent essentiellement à représenter les verbes ponts (= les verbes à complétive permettant l'extraction hors de celle-ci), partagent avec les arbres initiaux la propriété que l'ensemble des arguments sémantiques de l'ancre soient des arguments syntaxiques, tandis que les lexies représentées par un arbre modifieur ont un de leur argument sémantique comme gouverneur syntaxique. Comme nous le verrons dans la Section 4, le choix d'associer à une configuration un arbre prédicatif plutôt qu'un arbre initial est davantage liée à des contraintes procédurales (la procédure étant elle-même contrainte par la nature des opérations permettant de combiner les arbres entre eux) qu'à des propriétés linguistiques de la lexie concernée.

2.2. L'arbre de dérivation

Vue du point de vue de l'analyse, une phrase est reconnue par la grammaire TAG si on peut associer à chaque lexie pleine un arbre élémentaire de manière à ce que ces différents arbres élémentaires puissent être assemblés par substitution et adjonction en un arbre unique, appelé

l'arbre dérivé. Le témoin de cette analyse est appelé **l'arbre de dérivation**: chaque nœud de l'arbre de dérivation est le nom d'un arbre élémentaire et les branches de l'arbre se répartissent en deux types selon que l'arbre associé au nœud dépendant a été substitué ou adjoint à l'arbre associé au nœud gouverneur. Les branches de substitution sont représentées graphiquement par des traits pointillés et celles d'adjonction par des traits pleins.⁶

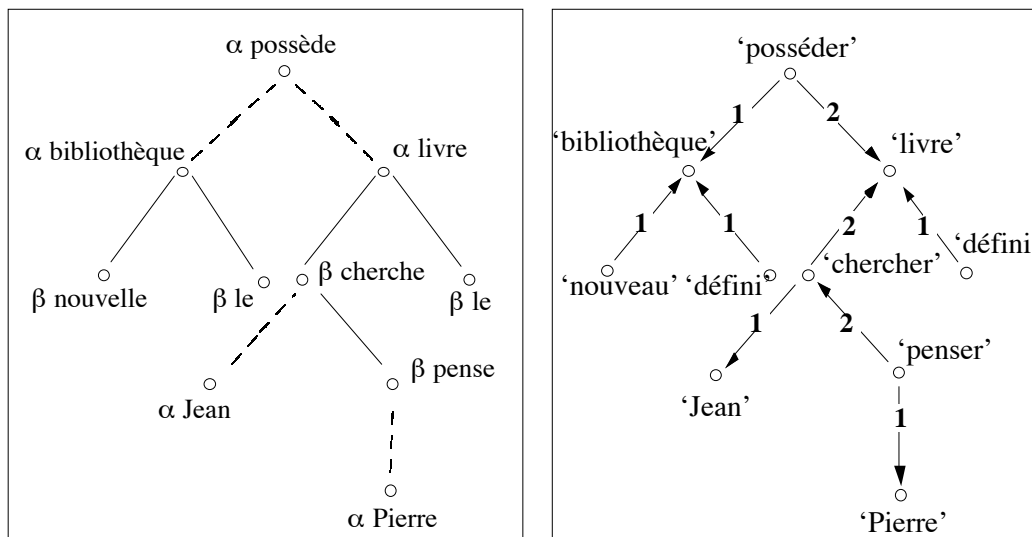


Figure 8 : l'arbre de dérivation et la SSém correspondante (disposée parallèlement)

3. Comparaison entre la SSém de la TST et l'arbre de dérivation

Dans cette section, nous établissons un parallèle entre SSém et arbre de dérivation. En TAG, l'arbre de dérivation encode totalement la façon dont est dérivée une phrase. On peut cependant le concevoir comme une représentation linguistique, indépendamment de la procédure qui donne l'arbre dérivé. Bien qu'il ait des propriétés hybrides entre syntaxe et sémantique, nous allons montrer qu'il existe un parallèle entre l'arbre de dérivation et la SSém en TST⁷. Prenons pour cela comme exemple l'arbre de dérivation et la SSém pour notre phrase de référence (Fig. 8). On peut faire les remarques suivantes :

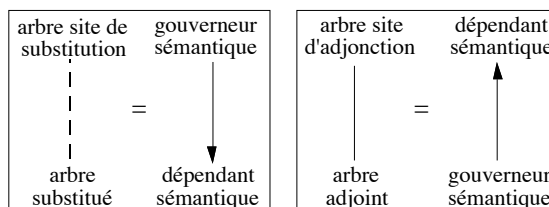
⁶ Nous prenons la définition de la dérivation TAG de (Schabes & Shieber 94). Pour être non ambiguë, la dérivation doit spécifier l'ordre dans lequel se font les adjonctions de différents arbres auxiliaires à un même nœud (le fils le plus à gauche est adjoint en premier et ainsi de suite). Un seul arbre prédictif peut être adjoint à un nœud donné, puisque l'ancrage de l'arbre prédictif devient après l'adjonction le gouverneur syntaxique de l'ancrage de l'arbre site. Comme le soulignaient déjà ces auteurs, cette définition de la dérivation est plus appropriée pour la représentation des dépendances sémantiques que celle où les branches ne sont pas ordonnées.

⁷ On pourrait être tenté de rapprocher l'arbre de dérivation de l'arbre SyntP plutôt que de la SSém (comme le font (Rambow & Joshi 94)) en raison de la structure d'arbre et de l'étiquetage par des lexies. Nous insistons sur le fait que les arcs de l'arbre de dérivation représentent bien des dépendances sémantiques. Comme il n'y a pas isomorphie entre dépendance sémantique et dépendance syntaxique (profonde ou de surface), il ne peut pas y avoir un parallèle systématique entre arbre de dérivation et SSyntP. Si nous reprenons la SSyntP de notre phrase de référence (Fig. 2), on peut voir plusieurs points de divergence avec la topologie de la SSém/arbre de dérivation. D'une manière générale, entre deux éléments en dépendance syntaxique profonde, il peut y avoir entre les sémantèmes correspondants (Melcuk, 88: 119sv):

- dépendance sémantique de même sens (par exemple entre 'posséder' et 'livre');
- dépendance sémantique de sens opposé (par exemple entre 'bibliothèque' et 'nouveau');
- pas de dépendance sémantique directe (par exemple entre 'livre' et 'penser').

Enfin il peut exister une dépendance sémantique directe entre deux sémantèmes dont les lexies ne sont pas liées directement par une dépendance syntaxique (comme, par exemple, entre 'boire' et 'vin', dans *boire un verre de vin*).

- Les nœuds de la SSém et de l'arbre de dérivation sont en bijection (cela est vrai sauf dans le cas de coréférences, où en TAG en général, on aura plusieurs nœuds dans l'arbre de dérivation, au lieu d'un dans la Rsém). Chaque nœud de l'arbre de dérivation renvoie à une unité sémantique i.e. un sémantème de la SSém, mais contient en outre des informations sur la lexicalisation de l'unité sémantique, la voix, la construction syntaxique, la pronominalisation, l'ordre linéaire, ...
- Chaque couple de sémantèmes reliés directement dans la SSém l'est aussi dans l'arbre de dérivation (c'est-à-dire que les deux représentations induisent un même graphe non orienté entre leurs nœuds)⁸.
- En ce qui concerne l'orientation des relations entre nœuds, chaque branche de substitution correspond à une dépendance sémantique de même sens, et chaque branche d'adjonction — modificative ou prédicative — correspond à une dépendance sémantique inversée.



Avec cette interprétation des arcs de substitution et d'adjonction, l'arbre de dérivation induit le même graphe *orienté* que la SSém.

4. Génération à partir d'une RSém avec TAG

Comme nous venons de montrer, il faut voir l'arbre de dérivation comme une SSém, à laquelle s'ajoutent les informations de niveau syntaxique et morphologique permettant la correspondance avec *une* phrase donnée. Une partie de ces informations est encodées par la SComm-Sém, mais une partie seulement, puisque la RSém (= SSém + SComm-Sém) représentent un invariant de paraphrase et peut généralement être exprimée par différentes phrases.

Pour générer une phrase à partir d'une RSém avec une grammaire TAG, le principe est simple: il faut passer de la RSém à un arbre de dérivation TAG (éventuellement sous-spécifié), puis effectuer la (ou les) dérivation(s) encodée(s) par cet arbre de dérivation.

La première étape consiste à choisir **un nœud d'entrée** pour la SSém, c'est-à-dire le nœud qui sera la racine de l'arbre de dérivation. Le choix de ce nœud dépend de la SComm-Sém et notamment des lexicalisations possibles des nœuds dominants des thème et rhème; de telles règles ont été proposées par (Iordanskaja et Polguère 88; Polguère 90)⁹.

⁸ Il existe des cas où cela n'est pas vrai. D'une part dans les cas de contrôle, la dépendance sémantique entre l'infinitif et son sujet n'est pas reflétée en TAG (à moins qu'un sujet vide soit substitué). D'autre part, les principes linguistiques qui président à la formation des arbres élémentaires (cf. §2.1) assurent que chaque nœud de l'arbre de dérivation contrôle autant de positions qu'il a d'arguments sémantiques. Néanmoins ces principes ne suffisent pas assurer que ces positions soient effectivement remplies par les arguments sémantiques, notamment à cause de l'adjonction prédicative. Ainsi pour dériver *Que Pierre veuille partir dérange Marie* en TAG, le sujet de *dérange* doit être substitué (pour bloquer l'extraction hors du sujet phrastique), mais comme *veuille* reste adjoint sur *partir*, l'arbre de dérivation possède un lien entre *dérange* et *partir* alors que c'est 'vouloir' l'argument sémantique de 'déranger' (*que Pierre veuille partir* exprime une volonté, pas un départ). (Rambow et al. 95) propose d'augmenter le pouvoir de l'opération de substitution afin de traiter les verbes ponts par des arbres initiaux et d'éviter ce type de problème (voir (Candito & Kahane, soumis) pour une étude détaillée).

⁹ Les règles de calcul du nœud d'entrée devront être adaptées aux spécificités de la grammaire TAG et notamment au fait que, en raison de leur traitement par des arbres auxiliaires, les verbes ponts ne peuvent pas être à la racine d'un arbre de dérivation.

Le graphe sémantique est ensuite **parcouru continûment** à partir du nœud d'entrée¹⁰; lorsqu'un arc est parcouru **positivement** — c'est-à-dire dans le sens gouverneur-dépendant —, on obtient un arc de substitution; lorsqu'un arc est parcouru **négativement** — c'est-à-dire dans le sens dépendant-gouverneur —, on obtient un arc d'adjonction (Cf. Fig 8).¹¹ Lors du parcours de la SSém, chaque sémantème reçoit en fonction notamment de la SComm-Sém des informations sur la lexicalisation (notamment la partie du discours), la diathèse, la construction syntaxique, etc, ainsi que des traits morphologiques traduisant les sémantèmes flexionnels pointant sur lui. Ceci donne un arbre de dérivation sous-spécifié dont les nœuds portent des traits (<passif>, <sujet_inversé>, ...) caractérisant un ensemble d'arbres élémentaires. On peut ensuite effectuer les différentes dérivations qu'encode cet arbre, comme cela est fait en G-TAG (Danlos & Meunier 96), et générer ainsi une ou plusieurs phrases.

Pour terminer nous allons voir que cette procédure peut en fait être appliquée à partir d'une grammaire Sens-Texte, puisque celle-ci peut être compilée pour donner une grammaire TAG.

5. Compilation d'une grammaire Sens-Texte en une grammaire TAG

Rappelons qu'à chaque entrée lexicale d'une grammaire TAG est associée une famille d'arbres élémentaires, chacun rassemblant des informations qui sont codées à différents niveaux en TST. Nous allons voir comment on peut construire une famille d'arbres élémentaires pour chaque entrée lexicale du dictionnaire de la TST, en utilisant les règles de correspondance Sens-Texte. Cela correspond formellement à une compilation de la grammaire Sens-Texte.

Pour construire à partir de l'entrée lexicale L et des règles de correspondance TST, la famille d'arbres élémentaires associée à L, l'idée est la suivante. On part de la configuration sémantique associée à L (c'est-à-dire le bout de graphe constitué de 'L' et de ses actants sémantiques 'X₁', ..., 'X_n'). En appliquant des règles de correspondance, on peut faire correspondre à cette configuration un bout d'arbre SyntP, puis un bout d'arbre SyntS et enfin une chaîne MorphP. Une séquence formée de la SSém associée à une lexie L, plus une SSyntP, une SSyntS et une SMorphS synthétisées est appelée une **chaîne de synthèse** pour L. En appliquant différentes règles de correspondance, on obtient différentes chaînes de synthèse pour L, dont chacune peut être traduite en un arbre syntagmatique élémentaire d'ancre L. On obtient ainsi la famille d'arbres élémentaires associée à L.

Partant d'une chaîne de synthèse, voyons comment obtenir un arbre élémentaire TAG.

Un arbre auxiliaire ou un arbre initial?

Les différentes chaînes de synthèse associées à L peuvent en particulier varier selon le parcours de la SSém associée à L : le parcours peut démarrer par 'L', mais aussi par chacun de ses dépendants.

- Si 'L' est le nœud par lequel commence la synthèse, tous les arcs sémantiques sont parcourus positivement. On obtient donc un **arbre initial** où les 'X_i' donnent les nœuds à substituer.

¹⁰ En raison des discontinuités entre la SSém et la SSyntP (cf. Note 7), il n'est pas possible, lors du passage de la SSém à la SSyntP d'assurer simultanément un parcours continu de la SSém et un traçage continu de la SSyntP. Dans (Kahane & Mel'cuk 97) est proposée une procédure où on privilégie un traçage continu de l'arbre SyntP du sommet vers les feuilles, quitte à effectuer des sauts dans le parcours de la SSém. Si on privilégie, comme ici un parcours continu de la SSém, on doit lors du traçage de l'arbre SyntP revenir en arrière insérer des bouts d'arbre, ce qui revient formellement à des adjonctions.

¹¹ Le parcours du graphe pose néanmoins un problème non négligeable (et non résolu) en raison de la présence de cycles: chaque cycle doit être coupé en un nœud. Le nœud où le cycle est coupé donne deux nœuds coréférents dont l'un des deux doit être pronominalisé.

- Sinon, la synthèse commence par un des actants sémantiques ' X_k '. L'arc pointant sur lui est parcouru négativement et tous les autres arcs sémantiques sont parcourus positivement. On obtient un **arbre auxiliaire** dont ' X_k ' donne le nœud pied et les autres ' X_i ' les nœuds à substituer.

Un arbre modifieur ou arbre prédicatif?

Si la synthèse commence par ' L ', L est nécessairement le sommet de l'arbre SyntS, Mais dans le cas contraire, i.e. dans le cas où on construit un arbre auxiliaire, L peut ou pas être la tête de l'arbre SyntP.

- Si X_k est le gouverneur SyntP de L , on obtient un **arbre modifieur**.
- Si L est le gouverneur SyntP de X_k , on obtient un **arbre prédicatif**.

Pour le reste le passage d'une chaîne de synthèse à un arbre élémentaire est relativement simple. L donne l'ancrage lexical de l'arbre élémentaire. Au dessus de L sont créés une ou plusieurs projections¹² auxquelles sont accrochés les nœuds à substituer, ainsi que le nœud pied dans le cas d'un arbre auxiliaire prédicatif. Dans le cas d'un arbre auxiliaire modifieur, la projection maximale de L et le nœud pied sont accrochés à la racine, qui est une projection du nœud pied.

Exemple 1: Considérons la chaîne de synthèse suivante (à gauche), obtenue à partir de l'entrée d'un verbe pont¹³:

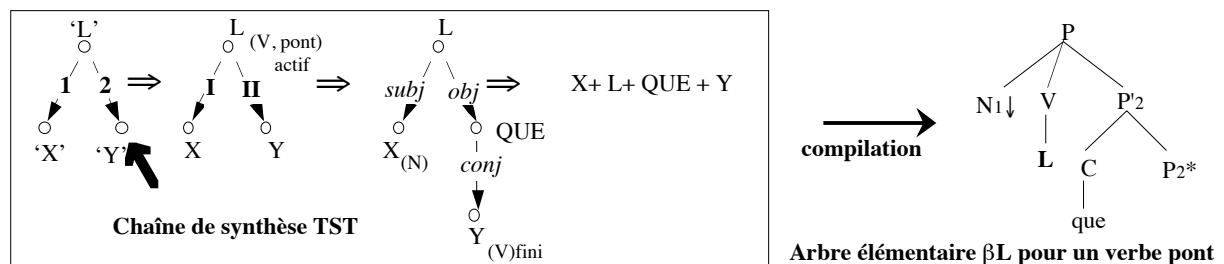


Figure 9 : Traduction d'une chaîne de synthèse TST en un arbre élémentaire TAG

Du fait que la synthèse commence en ' Y ' (indiqué par la grosse flèche), la chaîne de synthèse donne un arbre auxiliaire dont Y donne le nœud pied. Comme L est le gouverneur SyntS de Y , il s'agit d'un arbre auxiliaire prédicatif. L étant un verbe, un nœud V et un nœud P (projection maximale de V) sont placés au dessus de L . L'actant Sém 1 ' X ' donne le nœud à substituer N_1 placé à gauche de V ; l'actant Sém 2 ' Y ' donne le nœud pied de catégorie P_2 placé à droite de V . On obtient ainsi un arbre du type β pense (Fig. 7).

Conclusion

Notre étude n'est pas sans conséquences pour un modèle comme pour l'autre. Pour la TST, la « procédure de dérivation TAG » (c'est-à-dire le parcours continu de la SSém et la précompilation des règles de correspondance) fournit une procédure de référence pour toutes les études à venir sur les utilisations possibles de la TST en TAL. Même s'il est probable qu'une précompilation dans un format du type grammaires de dépendance serait plus économique, une opération du type

¹² Le nombre de projections de L est conséquence du nombre d'actants de L et de la nature des linéarisations possibles des actants de L et des adjoints en L . Nous ne traitons pas ce problème ici.

¹³ La chaîne de synthèse est obtenue comme suit. La synthèse commence en ' Y '. Bien que parcouru négativement, l'arc Sém 2 pointant sur ' Y ' peut être traduit par un arc SyntP II. Ceci est autorisé car L est un verbe pont et peut donc être inséré lors d'une extraction. En SyntS, une conjonction QUE est introduite. L'arc Sém 1, qui est parcouru positivement, donne une branche SyntP I, qui donne une branche SyntS *subjectale* dont le dépendant X est un nom; X est placé à gauche de L (et L s'accorde avec X).

de l'adjonction TAG devra être maintenue. De plus, le passage à une grammaire TAG reste à l'heure actuelle le meilleur moyen d'utiliser la TST pour l'analyse.

Pour TAG, il était clair avant cette étude (Vijay-Shanker & Schabes 92; Candito 96) qu'une grammaire TAG quelque peu exhaustive ne peut être écrite et maintenue sans une surcouche modulaire (comme l'est la grammaire TST). Même si la TST n'est peut-être pas le formalisme le mieux adapté pour synthétiser une grammaire TAG, elle peut être un guide précieux pour la réalisation d'une telle surcouche.

D'autre part, l'étude montre de manière précise qu'une dérivation TAG fournit davantage qu'une analyse syntaxique. Il s'agit d'une représentation sémantique (au sens de la TST), à la représentation de la coréférence près. Ce résultat a des applications évidentes pour le TAL utilisant les TAG, notamment la traduction automatique.

Références

- Abeillé A. (1991), *Une grammaire lexicalisée d'arbres adjoints pour le français*, Thèse de l'Univ. Paris 7.
- Candito M.-H. (1996), "A principled-based hierarchical representation of LTAG", *COLING'96*, Copenhagen, vol. 1, pp. 194-199.
- Danlos L. & Meunier F. (1996), "G-TAG, un formalisme pour la génération de textes : présentation et applications industrielles", *ILN'96*, Nantes.
- Franck R. (1992), *Syntactic locality and Tree Adjoining Grammar: Grammatical, Acquisition and Processing Perspectives*, Ph.D. Thesis Univ. of Pennsylvania.
- Kahane S. & Mel'cuk I. (1997), "Synthèse des phrases à extraction. Aspects sémantiques et syntaxiques", *CSSP'97*, Paris.
- Kroch A. (1989), "Asymmetries in Long-Distance Extraction in a TAG", in *Alternative conceptions of phrase structure*, Baltin & Kroch (eds), Univ. of Chicago Press.
- Kroch A. & Joshi A. (1985), "The linguistic relevance of tree adjoining grammar", Technical Report MS-CIS-85-18, Department of CIS, University of Pennsylvania.
- Iordanskaja L. & Polguère A. (1988), "Semantic processing for text generation", *ICSC'88*, Hong Kong, pp. 310-318.
- Mel'cuk I. (1988), *Dependency Syntax : Theory and Practice*, State Univ. Press NY, Albany.
- A. Polguère (1990), *Structuration et mise en jeu procédurale d'un modèle linguistique déclaratif dans un cadre de génération de texte*, Thèse de l'Univ. de Montréal.
- Rambow O. & Joshi A. (1994), "A formal look at dependency grammars and phrase-structure grammars, with special consideration of word-order phenomena", in Leo Wanner (ed) Pinter, London.
- Rambow, O., Vijay-Shanker, K. & Weir, D. (1995), "D-Tree Grammars", *ACL'95*.
- Schabes, Y. & Shieber, S. (1994), "An alternative conception of Tree Adjoining Derivation", *Computational Linguistics*, vol. 20, n° 1, pp. 91-123.
- Vijay-Shanker K & Schabes Y. (1992), "Structure sharing in a Lexicalized TAG". *COLING'92*, Nantes.
- XTAG Research Group (1995), "A Lexicalized TAG for English", Technical report IRCS 95-03, Univ. of Pennsylvania, (Online updated version).
- Zolkovskij A. & Mel'cuk I. (1967), "O semanticeskom sinteze [On semantic synthesis]", *Problemy kybernetiki*, v. 19, 177-238. [Trad. Franç : *T.A. Information* (1970), n°2, 1-85.]