[*Automatic Translation of Languages*: NATO Summer School...July 1962]

LANGAGES ARTIFICIELS, SYSTEMES FORMELS ET TRADUCTION AUTOMATIQUE†

BERNARD VAUQUOIS
Centre d'Etudes pour la Traduction Automatique, C.N.R.S., Faculte des Sciences, Grenoble

1. INTRODUCTION

POUR un linguiste ou, à plus forte raison encore, pour un sémanticien, l'étude des langues naturelles se présente principalement comme l'étude des moyens d'expression de la pensée. Aussi, le cheminement normal des recherches doit-il partir d'une analyse introspective des concepts et des relations entre ces concepts pour les projeter ensuite sur le moyen d'expression que constitue une langue naturelle. Les études du Professeur S. Ceccato dans ce domaine sont celles que nous connaissons le mieux et nous montrerons plus loin comment elles peuvent s'articuler avec d'autres manières d'envisager l'étude des langues (en vue de la traduction automatique). L'activité des linguistes structuralistes s'est détachée peu à peu de cette base purement sémantique pour se préoccuper surtout du moyen d'expression pour lui-même. Néanmoins, même dans les syntaxes structurales les plus formalisées, il persiste un appel implicite à la sémantique, appel qui apparaît sous plusieurs formes: en particulier, il semble que l'interprétation du système proposé précède la construction du système lui-même; ensuite, les schémas en arbre destinés à la représentation des structures syntaxiques peuvent présenter des ambiguités si l'on n'est pas guidé par un support sémantique; enfin, les schémas descriptifs procèdent par décomposition de la phrase et les critères formels sont insuffisants pour procéder à la composition d'une phrase en présence d'une suite d'éléments donnés.

Ces remarques ne constituent en rien une critique et nous utiliserons abondamment, par la suite, les travaux effectués dans ce domaine. Elles servent uniquement à introduire un autre point de vue, lié étroitement aux automates; c'est celui de la logique formelle dont le but est l'étude de langages artificiels considérés comme des assemblages de signes et que nous étendrons, dans la mesure du possible, à la représentation des langues naturelles. Cette dernière manière de voir s'applique directement aux langages écrits; dans le cas de langages parlés, il est nécessaire de passer à une forme écrite au moyen d'une notation phonétique.

En somme, si l'on place au niveau le plus haut le plan de la pensée (plan sémantique, si l'on veut) puis, en dessous, le plan de la langue naturelle (moyen d'expression) et enfin, au plus bas niveau, le plan des langages artificiels (assemblage de caractères), on voit que les études préalablement citées se situent aux deux niveaux les plus élevés et se tournent vers le niveau immédiatement inférieur. Ceci correspond à la démarche naturelle des études du point de vue "humain". Nous nous proposons alors d'étudier le niveau le plus bas et d'y incorporer progressivement les éléments des niveaux supérieurs. Ce "mouvement ascendant" s'adapte mieux au point de vue "machine". En effet, l'automate a la possibilité

† Presented at the NATO Advanced Study Institute on Automatic Translation of Languages, Venice, 15–31 July 1962.

de reconnaitre les caractères, les assemblages de caractères, et d'effectuer ainsi le transfert d'un texte écrit dans un langage artificiel vers son écriture dans un autre langage artificiel; par exemple, c'est le rôle d'un compilateur en programmation automatique.

Ainsi, nous sommes naturellement conduits à envisager la traduction automatique comme le passage entre deux langages artificiels, le premier étant un modèle de la langue naturelle source, le deuxième représentant la langue cible. A ce problème de transfert, dont la résolution peut s'inspirer dans une certaine mesure des techniques de compilation, s'ajoute celui de la fabrication des modèles.

Au cours de cette première leçon, nous nous bornerons à une étude sommaire des langages artificiels; ceci signifie que nous nous contenterons d'énoncer des concepts sans entrer dans des détails trop compliqués et sans faire le tour des nombreuses applications.

2. LANGAGES ARTIFICIELS ET SYSTEMES FORMELS

2.1. *Concepts de base*

En premier lieu, toute référence à un domaine de signification extérieur au système est éliminée. L'intuition ne porte plus sur le contenu des expressions; elle se borne au fait de pouvoir identifier un signe, distinguer deux signes différents, remplacer un signe par un autre suivant un modèle de substitution [1], [2], [3].

En conséquence, les premières données sont des "*symboles formels*" c'est-à-dire *un ensemble de caractères* appelé "*alphabet*". Dans le cas où plusieurs symboles de typographie différente seraient considérés comme représentatifs d'un même caractère, il convient de distinguer le *niveau graphétique* du *niveau graphémique* [4]. Ainsi, au niveau graphétique, le même caractère peut apparaitre sous des dessins différents; par contre, au niveau graphémique, on ne rencontre que des caractères différents au sens du langage. Nous nous plaçons à ce deuxième niveau.

Ensuite, apparaît une opération de base: c'est la "*concaténation*". Cette opération permet de construire des "*expressions formelles*" à partir des symboles formels (caractères) ou d'expressions formelles plus simples qui deviennent alors des "*sous-expression*". Par exemple si "α, β, γ, δ" constituent un alphabet

$$\begin{cases}\beta \\ \alpha\gamma\gamma\delta\end{cases} \quad \text{constituent des expressions.}$$

Une autre opération de base est la "*substitution*".

Elle consiste à substituer une expression donnée aux occurrences d'une sous-expression appartenant à une expression.

Dans l'expression $\alpha\gamma\gamma\delta$, γ peut être considérée comme une sous-expression; si on lui substitue l'expression $\beta\beta$, on obtient l'expression: $\alpha\beta\beta\beta\beta\delta$.

On peut considérer alors l'ensemble $\mathscr{E}$ de toutes les expressions qu'il est possible de construire avec un alphabet donné. Dans ces conditions, définir un langage artificiel $\mathscr{L}$ revient à déterminer un sous-ensemble de $\mathscr{E}$ [5]. Le langage ainsi défini, doit permettre la résolution des deux problèmes suivants:

(*a*) Etant donné l'alphabet, construire toutes les expressions du langage (problème de construction).

(*b*) Etant donnée une expression compatible avec l'alphabet, décider si cette expression appartient au langage ou non. (Problème de reconnaissance.)

2.2. *Description des langages*

Les deux problèmes précédents sont évidemment liés, mais nous commencerons par l'examen du premier parce qu'il est plus facile. Pour réaliser la construction de toutes les expressions du langage il est nécessaire de distinguer les concaténations et substitutions qui sont permises de celles qui ne le sont pas. Ceci s'obtient par la donnée de "*règles de constructions*" de $\mathscr{L}$. Mais, pour exprimer ces règles il faudra utiliser un certain langage $\mathscr{M}$.

$\mathscr{M}$ est dite "*métalangue*" pour le langage $\mathscr{L}$, qui est lui-même par rapport à $\mathscr{M}$, *le* "*langage objet*". Lorsque le langage objet est une langue naturelle, le plus souvent la métalangue est constituée par cette même langue naturelle et ceci peut sembler paradoxal.

Si le langage objet est un langage artificiel, la métalangue en est un autre et l'on se heurte aussi à une difficulté du même ordre; en effet, il faudrait une méta-métalangue pour décrire la métalangue et ainsi de suite. En réalité, ces difficultés disparaissent si l'on se rappelle que l'élément à formaliser est le langage $\mathscr{L}$. Ainsi, dans le premier cas, par formalisations successives on peut décrire une langue naturelle avec de plus en plus de rigueur; dans le second cas, description d'un langage artificiel, on peut s'aider d'une métalangue décrite simplement; ce procédé à l'avantage d'être extrèmement précis; malheureusement, une métalangue simple n'est pas toujours un outil assez puissant pour décrire complètement $\mathscr{L}$. Aussi, en cas de nécessité fera-t-on appel à l'emploi d'une langue naturelle dans le rôle de métalangue, en gardant le souci de la plus grande précision.

A titre d'exemple, la métalangue simple donnée dans le rapport ALGOL 60 est une extension d'un système proposé par J. Backus [6].

Cette métalangue comprend les 4 symboles suivants:

$$::= |, \langle, \rangle$$

Les crochets $\langle\ \rangle$ servent à contenir des variables métalinguistiques qui se rapportent au langage $\mathscr{L}$ et en représentent des expressions ou sous-expressions; le symbole $::=$ est une connective de définition et le symbole | est une connective d'énumération.

Ainsi, si l'alphabet de $\mathscr{L}$ se compose des signes: $A\,B\,C\,D12$, on pourra définir:

$$\langle\text{lettre}\rangle ::= A|B|C|D$$
$$\langle\text{chiffre}\rangle ::= 1|2$$

Au moyen de "*définitions récursives*" on pourra introduire certains types d'expression, telles que:

$$\langle\text{identificateur}\rangle ::= \langle\text{lettre}\rangle|\langle\text{identificateur}\rangle\langle\text{lettre}\rangle|\langle\text{identificateur}\rangle\langle\text{chiffre}\rangle$$

Ainsi l'on vient de voir comment on peut exprimer les règles de construction des expressions d'un langage $\mathscr{L}$. En ce qui concerne le problème de la reconnaissance on pourra utiliser ces mêmes règles pour déterminer lesquelles s'appliquent à la construction de l'expression donnée à priori. Dans le cas où aucune règle ne peut y parvenir, alors il s'agit d'une expression extérieure au langage.

2.3. *Eléments d'un langage artificiel*

Dans l'exemple précédent, nous avons distingué deux types de caractères dans l'alphabet (type "lettre" et type "chiffre"). Il va de soi qu'un langage artificiel disposant d'un alphabet aussi pauvre en types de caractères ne permet pas d'aller bien loin. Sans quitter la généralité ou nous nous sommes placés, c'est-à-dire sans entrer dans les conventions particulières à tel ou tel langage, nous pouvons dégager un certain nombre d'éléments que l'on rencontre à peu près partout.

(*a*) *Des variables.* On désigne sous ce vocable des éléments formels susceptibles de prendre certaines valeurs; ces valeurs sont explicitées sur le plan de l'interprétation que nous verrons plus loin. Sur le plan formel, ces variables peuvent être des symboles de l'alphabet ou bien des expressions obtenues à partir de symboles de base au moyen de règles de construction propres au langage $\mathscr{L}$. (Tel, par example, l'indentificateur énoncé au paragraphe précédent.)

(*b*) *Des constantes.* Il s'agit ici de symboles formels qui représentent directement des valeurs que peuvent prendre les variables. Si, par exemple, les variables sont du type booléen, on pourra introduire les deux constantes représentatives respectivement des valeurs "vrai" ou "faux".

(*c*) *Des séparateurs.* Ce sont des symboles de l'alphabet qui permettent de séparer deux éléments distincts lorsque leur concaténation donnerait lieu à une ambiguité. Le "blanc" ou "espace" est un séparateur.

(*d*) *Des opérateurs.* Ils sont aussi représentés par des symboles de l'alphabet. Ils servent à mettre en relation deux éléments du type "variable" ou "constante" ou encore des expressions plus compliquées formées à partir de ces premiers éléments. On est alors en présence d'opérateurs binaires. Certains opérateurs ne portent que sur une seule expression. On les appellera "opérateurs 1-aire" (par exemple l'opérateur de négation). Le rôle des opérateurs est précisé par les règles de construction du langage.

A la notion d'opérateur s'adjoint celle de "*portée d'un opérateur*". Un opérateur 1-aire n'a qu'une seule portée, en général à droite. Par exemple, si A est une variable et $\neg$ un opérateur 1-aire $\neg A$ est une expression dans laquelle la portée de $\neg$ est A.

Dans le cas d'un opérateur binaire, il peut exister une portée à droite et une portée à gauche. Si par exemple, A et B sont des variables et $\perp$ un opérateur binaire.

$A \perp B$ est une expression dans laquelle A et B sont respectivement les portées à gauche et à droite de $\perp$.

Un opérateur binaire peut n'avoir qu'une portée (à droite) si l'on prend la convention de l'écriture préfixée.

Ainsi dans, $\perp AB$, $\perp$ ne présente qu'une portée à droite, A et B étant les deux termes soumis à l'opérateur.

Si un opérateur s'adjoint une variable figurant dans une expression contenue dans sa portée, cette variable est dite "liée"; toute variable non liée est dite "libre".

(*e*) *Des parenthèses.* Ce sont des symboles, donnés par paire (une parenthèse ouvrante et une parenthèse fermante) qui servent en général à établir des hiérarchies à l'intérieur des expressions. Ces hiérarchies sont purement formelles et peuvent être respectées ou non sur le plan de l'interprétation. Exemples de parenthèses: (), [], ⟨⟩, {}, etc.

On a également l'habitude de distinguer dans un langage artificiel, divers types d'expressions, suivant le groupe de règles utilisé pour les former. Ainsi, au delà des symboles (éléments de l'alphabet) fait-on intervenir des "*termes*", et, à un niveau de complexité supérieur, des "*formules*".

2.4. *Systèmes formels*

Une particularisation interessante de la notion de langage artificiel est celle de "*système formel*". Bien que cette notion n'interesse pas directement la traduction automatique il est utile de l'introduire ici afin de distinguer la différence, sur le plan de l'interprétation, entre système formel, modèle d'une théorie mathématique d'une part, et langage artificiel, modèle d'une langue naturelle d'autre part.

Nous avons dit qu'à partir d'un alphabet, on pouvait envisager l'ensemble $\mathscr{E}$ de toutes les expressions compatibles avec cet alphabet et qu'un langage $\mathscr{L}$ est un sous-ensemble de $\mathscr{E}$. $\mathscr{L}$ est donc un certain ensemble d'expressions que l'on sait construire au moyen des règles de constructions propres à ce langage; nous avons dit que certaines d'entre elles, que l'on sait reconnaitre, sont appelées "formules". Si, maintenant, on choisit quelques unes de ces formules, qu'on appellera "formules valables" (ou "axiomes") et si on ajoute aux règles de construction formelle des règles de déduction qui permettent à partir de formules valables d'obtenir d'autres formules valables, on définit alors un "*système formel*".

Ainsi dans un système formel, aux problèmes de construction et de reconnaissance des expressions appartenant au langage, s'ajoutent les problèmes de construction et de reconnaissance des formules valables.

3. INTERPRETATION

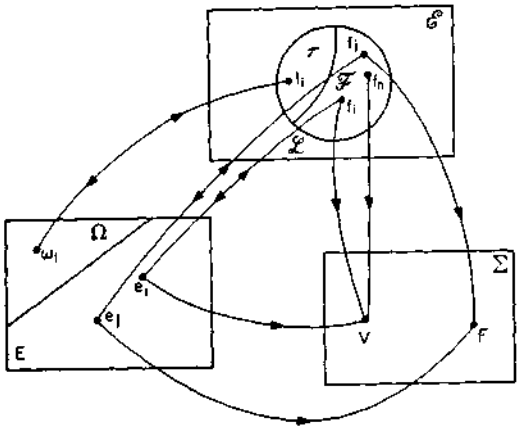
3.1. *Représentation et Interprétation d'un système formel*

Nous introduisons les notions de représentation et d'interprétation d'abord à propos des systèmes formels, car bien que cela conduise à des problèmes assez difficiles, ces problèmes sont beaucoup moins compliqués que ceux qui apparaitront lors de la généralisation de ces notions aux langages artificiels représentatifs de langues naturelles. Ainsi, étant donné un système formel, on est capable de réaliser la construction formelle de termes. Si l'on établit une correspondance biunivoque entre les termes du système et les éléments d'une classe d'objets donnée par ailleurs, on définit alors une "*représentation*" du système. Un même système est évidemment susceptible de recevoir plusieurs représentations distinctes. Une représentation est ainsi une concrétisation du système formel.

Faisons maintenant appel à des concepts sémantiques le plus simpiement possible. Nous introduisons alors les notions complémentaires de "*vrai*" et "*faux*". Supposons maintenant que les formules du système soient susceptibles de prendre deux valeurs seulement (des propositions, par exemple). Si on établit une correspondance entre les formules du système et une certaine classe d'énoncés dont les valeurs "vrai" ou "faux" sont déterminées en dehors du système et si, en outre, les formules valables du système et celles qu'on peut déduire correspondent à des éconcés vrais, on a alors défini une "*interprétation*" du système formel.

Ce qui précède peut encore être schématisé de la façon suivante: Soient $\mathscr{E}$ l'ensemble des expressions compatibles avec un alphabet et $\mathscr{L} \subset \mathscr{E}$ un langage artificiel. Considérons

FIG. 1.



dans $\mathscr{L}$ le sous-ensemble τ des termes et celui $\mathscr{F}$ des formules. Le système formel fournit les éléments $f_1, f_2, \ldots f_n$ de $\mathscr{F}$ qui sont "valables". Soit par ailleurs un champ sémantique de concepts $\sum$ dont les deux seuls éléments qui nous interessent sont le concept "vrai" (V) et le concept "faux" (F). Enfin, considérons un ensemble d'objets Ω et un ensemble d'énoncés E que l'on peut former au sujet de ces objets. La figure 1 montre ce qu'est la représentation et l'interprétation du système formel.

Il convient de noter que le système ayant un opérateur de négation $\neg$, on déduit la valeur "faux" d'une formule f_j si l'on est capable de déduire la valeur "vrai" de $\neg f_j$.

Dans ces conditions une formule $f \in \mathscr{F}$ peut être toujours mise en correspondance avec un énoncé $e \in E$, mais on montre qu'il n'est pas toujours possible de déterminer la valeur V ou F par application dans $\sum$; c'est le cas des formules indécidables. Lorsque nous augmenterons considérablement le nombre d'éléments de $\sum$ mis en jeu pour obtenir une interprétation satisfaisante langage artificiel-langue naturelle, il faudra donc nous attendre à des difficultés encore plus grandes.

3.2. *Interprétation des langages artificiels—Schéma sommaire*

3.2.1. *Langages artificiels, modèles de langues naturelles.* A partir de ce que nous venons de voir sur la représentation et l'interprétation d'un système formel, nous allons concrétiser ces notions en les appliquant aux langues naturelles; il nous sera nécéssaire d'enrichir considérablement l'ensemble des concepts sémantiques mis en jeu pour obtenir une interprétation satisfaisante, et c'est dans ce domaine que nous articulerons cette présentation formaliste des langages avec les travaux cités à l'introduction. Nous sautons ainsi l'étape intermédiaire qui consiste à étudier les langages artificiels en tant que langage de programmation (où l'ensemble sémantique est plus réduit) et l'on peut se référer sur ce sujet à l'article de F. Genuys [5].

Une première différence, importante, est la suivante: au niveau de la représentation d'un système formel les correspondances entre termes (t_i) et objets (w_i) d'une part et entre formules (f_i) et énoncés (e_i) d'autre part sont des correspondances biunivoques; maintenant, si les objets deviennent des mots (au sens habituel) et si les énoncés devienennt des phrases d'une langue naturelle, les correspondances précédentes ne sont plus biuni

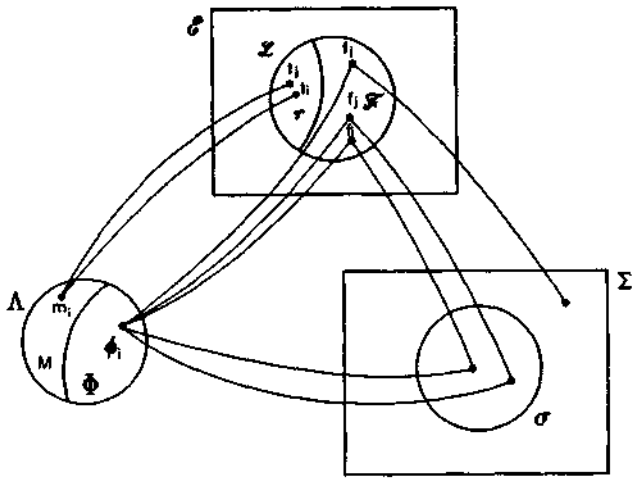


FIG. 2. A: langue naturelle, représentation de $\mathscr{L}$, M: ensemble des mots de A. m_i: élément de M.

voques. Ensuite une deuxième différence, source d'obstacles plus difficiles à surmonter, apparaît au niveau de l'interprétation: les projections faites sur le plan sémantique ne visent plus seulement des concepts isolés et simples comme "vrai" et "faux" mais concernent aussi des éléments obtenus par des relations entre concepts et ici encore, les correspondances ne sont plus biunivoques.

On obtient alors un schéma sommaire qui à l'allure représentée en figure 2.

ϕ: ensemble des phrases de Λ
φ_i: élément de ϕ
$\mathscr{L}$: langage artificiel
τ: ensemble des termes de $\mathscr{L}$; t élément de τ
$\mathscr{F}$: ensemble des formules de $\mathscr{L}$; f élément de $\mathscr{F}$
$\sum$: ensemble des concepts et des éléments obtenus au moyen de relations entre les concepts
σ: sous-ensemble des éléments précédents considérés comme valables.

3.2.2. *Liaison avec les problèmes de traduction automatique.* En étendant le schéma précédent au cas de deux langages, on met facilement en évidence les diverses conceptions que l'on peut imaginer de la traduction automatique (figure 3).

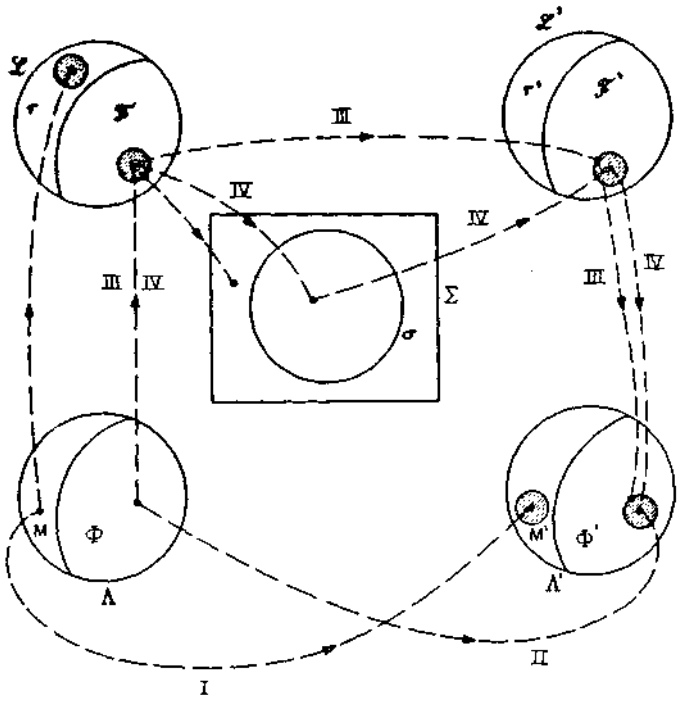


FIG. 3.

Le chemin 1 correspond à la traduction mot à mot de la langue naturelle Λ vers la langue naturelle Λ'.

Ce procédé ne présente que peu d'intérêt puisqu'à un élément de M correspondent en général plusieurs éléments de M' (symbolisés par une plage hachurée sur la figure).

Le chemin 2 correspond à la traduction directe, phrase à phrase de Λ vers Λ'. Le résultat est unique, mais si l'ensemble M est fini, les ensembles ϕ et ϕ' sont par contre infinis. Il est donc exclu de trouver une solution par ce procédé.

Le chemin 3 passe par l'intermédiaire des langages artificiels $\mathscr{L}$ et $\mathscr{L}'$ mais ignore le niveau de l'interprétation.

Les difficiles problèmes de décision, quant à la validité des formules sont ainsi évités; en contre-partie, on aboutit à une multiplicité des solutions dans Λ.

Enfin le chemin 4 passant par le niveau de l'interprétation est sans doute le plus long et le plus difficile; il a l'avantage de conduire à une solution unique dans la plupart des cas et éventuellement à une multiplicité de solutions extrêmement réduite.

Il va de soi que la figure 3 est un schéma très sommaire. Il se contente d'illustrer la progression réalisée depuis les notions de langages artificiels et de systèmes formels. L'examen plus détaillé des différentes composantes de ce schéma fera l'objet des leçons suivantes.

4. OBJECTIFS D'UN LANGAGE ARTIFICIEL, MODELE D'UNE LANGUE NATURELLE

Dans ce qui suit nous désignerons par $\mathscr{L}$ le langage artificiel et par Λ la langue naturelle qui en est la représentation. Comme les problèmes de traduction mettent en jeu au moins deux langues naturelles, nous avons intérêt à conserver le plus longtemps possible les éléments susceptibles du plus grand nombre de représentations. Ainsi en quittant le plan des généralités sur les langages de la leçon précédente pour particulariser jusqu'à la construction de modèles de langues naturelles, nous nous limiterons au strict nécessaire. L'avantage de cette particularisation restreinte ne se borne d'ailleurs pas à la possibilité d'une représentation multiple de langues naturelles, mais permet également d'interpréter des théories linguistiques d'apparence différente.

Lorsqu'on applique systèmatiquement une théorie linguistique à une langue particulière, on constate que certains faits de cette langue sont en désaccord avec la théorie. Ceci n'a rien de surprenant car un modèle n'est qu'une approximation de la réalité. Mais, cette cause peut ne pas être la seule et les désaccords constatés dans de nombreux cas proviennent du fait que la théorie laisse une part trop importante à l'intuition. Aussi, la construction du langage artificiel présente-elle l'avantage de faire préciser rigoureusement certains aspects des théories linguistiques.

5. ELEMENTS DE BASE DU MODELE

5.1. *Le choix des termes*

Comme le niveau le plus bas dans un langage est celui des caractères la question qui se pose est de savoir si les symboles formels de base sont susceptibles d'une représentation dans une langue naturelle au moins. Si la réponse est positive (ou du moins, partiellement positive) pour des langues telles que le chinois ou le japonais, elle est négative pour les langues indo-européennes qui sont les principales représentations des modèles que nous cherchons à construire. En effet, en russe, en anglais, en allemand, en français, etc., les symboles de l'alphabet ne représentent rien (à l'exception bien sûr des signes de ponctuation et des chiffres dont nous parlerons plus tard). Les éléments de plus bas niveau qui soient à la fois discernables automatiquement et susceptibles d'une représentation dans ces langues, sont certaines chaines de symboles limitées par des séparateurs (en général le "blanc", éventuellement des signes de ponctuation). La représentation d'une telle chaine dans une

langue naturelle correspond à ce qu'on appelle communément "*mot*". D'une manière plus précise nous appellerons "*forme*" toute chaine de symboles comprise entre deux séparateurs dans une langue naturelle.

(Ceci correspond à ce que nous avions appelé "terme primaire" dans une précédente communication [7]); et nous entendrons par "mot" tout ce que représente sa "forme" pour un témoin de la langue. Par exemple, le mot:

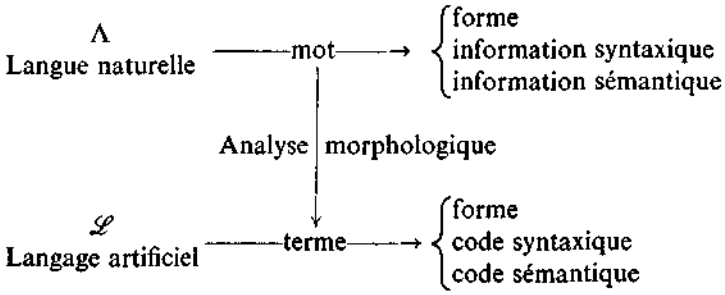
ASPIRATEUR

qui, sur le papier, et hors de tout contexte, donne naissance à une forme (la suite de lettre *A*, *S*, *P*, etc.) qui est une quantité d'informations de nature formelle, évoque aussi un élément du langage susceptible de certaines combinaisons (formation du pluriel avec un *S*, propriétés d'associations caractérisant la classe "substantif", etc.) et surtout un certain appareil electro-ménager. Ces deux dernières informations, respectivement de nature syntaxique et sémantique n'apparaissent pas de manière formelle dans la langue naturelle; ou, dans le langage artificiel modèle il faut tout formaliser. En conséquence, dans ce langage, la forme doit être accompagnée d'un certain nombre de codes fournissant ainsi une information identique à celle contenue dans le "mot". Il s'agit là du modèle parfait, probablement irréalisable, et il faut s'attendre à ce que les codes ne donnent qu'une partie de cette information.

Ces formes accompagnées de leurs codes sont les "termes" de notre système; leurs représentations sont les mots de la langue. L'élément commun qui permet de passer de Λ à $\mathscr{L}$ est la forme seule.

Nous appellerons "*analyse morphologique*" l'opération qui consiste à passer du mot au terme (éventuellement à plusieurs termes).

Nous en déduisons le schéme suivant:



On peut remarquer que le choix des termes peut se fonder sur d'autres critères comme par exemple celui de la coïncidence maximum entre la chaine du texte en langue naturelle et un dictionnaire d'éléments comprenant aussi bien des formes correspondant à des mots complets que des parties de mots ou des locutions composées de plusieurs mots. Dans ce qui suit, nous nous en tiendrons à notre premier choix et le terme du système $\mathscr{L}$ sera défini par:

$$\langle \text{terme} \rangle :: = \langle \text{forme} \rangle \langle \text{code syntaxique} \rangle \langle \text{code sémantique} \rangle$$

5.2. *Les éléments du code syntaxique*

Considérons dans la langue naturelle Λ l'ensemble des formes. Soit d'autre part un dictionnaire de cette langue; on y trouve non pas toutes les formes mais seulement certaines d'entre-elles, chacune représentant un sous-ensemble (forme du substantif au nominatif

singulier, du verbe à l'infinif, etc.). Chacun de ces sous-ensembles, caractérisé dans le dictionnaire par l'un de ses éléments, sera appelé "*unité lexicale*" (U.L.).

Enfin, dans un très grand nombre de langues, il existe des "*variables grammaticales*" (*v.g.*), chacune d'elles étant susceptible de prendre un certain nombre de valeurs. Ainsi, étant donnée la liste des variables grammaticales $[vg_1, \ldots vg_n]$ on peut faire une partition de l'ensemble des formes par rapport à ces variables grammaticales (figure 4). Choisissons,

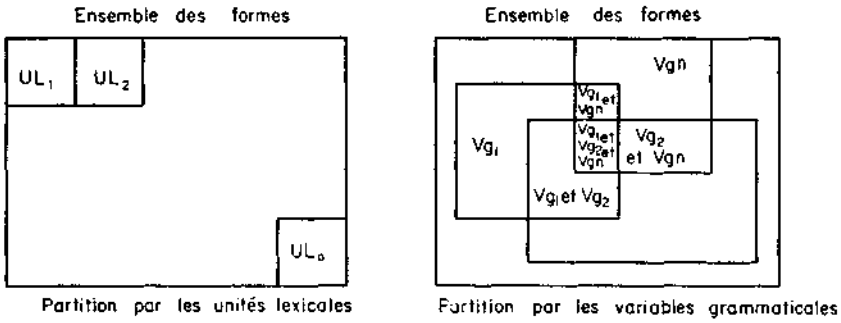


FIG. 4.

dans l'ensemble des formes, certaines d'entre elles relevant du même groupe de variables grammaticales soit par exemple:

$$vg_i,\ vg_j,\ vg_k.$$

Si l'on connait l'unité lexicale dont dépend une forme, cette unité lexicale peut fournir la valeur (éventuellement plusieurs) de certaines *v.g.*; les valeurs prises par les autres *v.g.* ne sont connues qu'au niveau de la forme. Nous appellerons les premières "variables grammaticales permanentes" (*v.g.p.*) et les secondes "variables grammaticales contingentes" (*v.g.c.*).

Exemple: Soient en français, les variables grammaticales.

GENRE dont les valeurs possibles sont Masculin et Féminin

et

NOMBRE dont les valeurs possibles sont au Singulier et Pluriel.

Choisissons les formes "COURTES" et "RENARD".

La première appartient à une certaine unité lexicale UL (court) qui comprend les 4 formes "COURT", "COURTE", "COURTS" et "COURTES", la deuxième à UL (RENARD) qui comprend les deux formes "RENARD" et "RENARDS".

La donnée de l'UL (COURT) ne précise la valeur ni du genre ni du nombre; il faut aller au niveau de la forme "COURTES" pour obtenir

GENRE Féminin
NOMBRE Pluriel

Par contre la donnée de UL (RENARD) détermine GENRE masculin et la forme "RENARD" indique NOMBRE singulier.

On dira que l'unité lexicale (COURT) possède les *v.g.* "GENRE" et "NOMBRE" en tant que contingentes alors que l'U.L. (RENARD) possède la *v.g.* "GENRE" en tant que permanente et la *v.g.* "NOMBRE" en tant que contingente.

En tenant compte de la nature (permanente ou contingente) des variables on obtient une partition plus fine de l'ensemble des formes. Ainsi, dans l'exemple précédent, l'U.L. (COURT) et l'U.L. (RENARD) se trouvent dans deux sous-ensembles distincts.

Provisoirement appelons "*catégorie syntaxique*" K un tel sous-ensemble. Remarquons que toutes les formes d'une même unité lexicale ne font pas obligatoirement partie de la même catégorie syntaxique. Ainsi, certaines formes liées à un verbe auront les *v.g.* "PERSONNE, NOMBRE, TEMPS, MODE" alors que d'autres auront les *v.g.* "GENRE, NOMBRE" et que d'autres enfin n'auront aucune *v.g.* (figure 5).

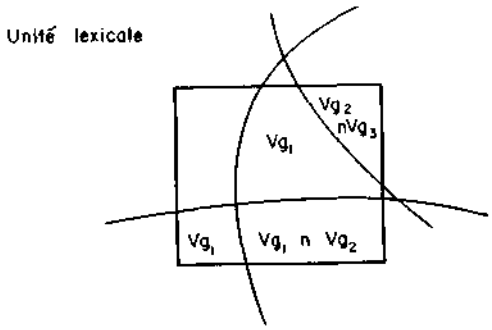


FIG. 5.

Nous préciserons plus loin la définition des catégories syntaxiques; nous retiendrons seulement le fait que dans le code syntaxique devront figurer les variables K, *vgp* et *vgc*.

Si en outre, on rappelle que le code syntaxique d'un terme doit donner les informations relatives aux liaisons syntaxiques particulières du mot associé, on est conduit à prévoir un code spécifiant le gouvernement et un code spécifiant la dépendance de ce terme par rapport à d'autres termes. Nous appellerons *c.g.* et *c.d.* ces deux derniers codes.

Ainsi l'on obtient:

$$\langle \text{code syntaxique} \rangle :: = \langle K \rangle \langle vgp \rangle \langle vgc \rangle \langle cg \rangle \langle cd \rangle$$

5.3. *Les opérateurs syntaxiques*

Nous avons vu, au cours de la première leçon, que parmi les éléments de tout langage artificiel, figuraient des opérateurs. Dans tous les langages artificiels dont on est maître de la construction, on donne des règles de formation des expressions telles que les opérateurs y apparaissent explicitement, que la portée ou les portées de ces opérateurs ne soient pas ambigues et enfin que ces portées soient adjacentes aux opérateurs. Pour définir les portées de manière univoque on dispose alors des parenthèses et, si on le désire, on établit des règles de priorité entre les opérateurs. Ainsi, par exemple, en considérant des expressions arithmétiques contenant les opérateurs $+$ et $\times$ et en donnant la priorité à ce dernier, dans l'expression

$$A \times B + C \times D$$

la portée de l'opérateur $+$ est $A \times B$ à gauche et $C \times D$ à droite.

Dans l'expression $A + B \times C + D$, la portée de $\times$ est B à gauche et C à droite. Si l'on veut avoir $A + B$ comme portée à gauche et $C + D$ comme portée à droite, il faut alors écrire l'expression:

$$(A + B) \times (C + D)$$

Dans le langage artificiel que nous voulons construire nous introduisons également des opérateurs; comme ces derniers ont pour représentateur les liaisons syntaxiques dans une langue naturelle, nous les nommons "*opérateurs syntaxiques*" ou encore "*fonctions syntaxiques*" φ. Reprenons le terme du langage artificiel

$$\langle\text{terme}\rangle :: = \langle\text{forme}\rangle\langle\text{code syntaxique}\rangle\langle\text{code sémantique}\rangle$$

Nous pouvons faire une opération de substitution qui consiste à remplacer la forme par le numéro d'unité lexicale à laquelle elle appartient. Ensuite, dans un texte de la langue naturelle, le mot apparaît à une certaine place; l'information relative à cette place est indiquée au moyen d'un numéro séquentiel v. Nous considérons alors le terme ordonné défini par

$$\langle\text{terme ordonné}\rangle :: = \langle v\rangle\langle UL\rangle\langle\text{code syntaxique}\rangle\langle\text{code sémantique}\rangle.$$

Nous appelons "syntagme élémentaire" (*s.e.*) le terme ordonné privé se son code sémantique. Dans ces conditions, nous avons:

$$\langle s.e.\rangle :: = \langle v\rangle\langle UL\rangle\langle K\rangle\langle Vgp\rangle\langle Vgc\rangle\langle Cg\rangle\langle Cd\rangle.$$

A une même forme peuvent correspondre plusieurs syntagmes élémentaires et nous allons en donner une classification [8], [9]. Soit f la forme et considérons un syntagme élémentaire $s.e._1$ déduit de cette forme.

Si $s.e._2$ est un autre syntagme élémentaire déduit de f alors:

(*a*) $v_1 = v_2$ de toute évidence. Si $UL_1 = UL_2$, $K_1 = K_2$, $vgp_1 = vgp_2$, $cg_1 = cg_2$ et $cd_1 = cd_2$ de sorte que seul $vgc_1 \neq vgc_2$, nous dirons alors qu'il s'agit "*d'homographes internes*".

Exemple: GAZ $\begin{cases} vgc_1 : \text{nombre singulier} \\ vgc_2 : \text{nombre pluriel.} \end{cases}$

(*b*) $UL_1 \neq UL_2$ et $K_1 \neq K_2$: alors les variables grammaticales ne sont plus comparables. Nous dirons dans ce cas qu'il s'agit "*d'homographes externes*".

Exemple: SORT $\begin{cases} UL_1\ K_1 : \text{substantif} \\ UL_2\ K_2 : \text{verbe.} \end{cases}$

Le cas d'homographie externe n'exclut pas celui d'homographie interne:
Exemple: BOIS, FERME, etc.

(*c*) Nous appellerons "polysémies" les termes ordonnés dont la partie "code sémantique comprend plusieurs unités sémantiques.

Le traitement des polysémies est hors du propos de l'analyse syntaxique.
Exemple: FACTEUR.

(*d*) $UL_1 \neq UL_2$ et $vgp_1 \neq vgp_2$.

Alors selon les règles de combinaison que le contexte permet on se ramène soit au cas des homographies externes soit à celui des polysémies.
Exemple: MANCHE, MODE, etc.

Soient deux syntagmes élémentaires s_i et s_j tels que $vs_i \neq vs_j$, une fonction syntaxique φ_p pourra être utilisée pour créer un "syntagme" s_k

$$\varphi_p(s_i, s_j) = s_k.$$

Au nouveau syntagme s_k on associera un numéro de niveau, et en donnant le numéro de niveau 0 aux syntagmes élémentaires ceux-ci deviennent des cas particuliers de "syntagmes". On peut alors envisager toutes les fonctions φ_p qui portent sur deux syntagmes.

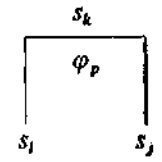
6. LES REGLES DE CONSTRUCTION SYNTAXIQUE

6.1. *Construction et interprétation des syntagmes*

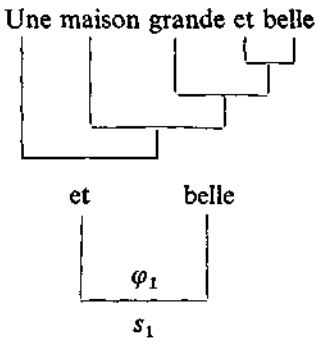
Rapportons nous à la figure 3 de la première leçon. Il importe de détailler davantage la structure de $\mathscr{L}$. Au lieu de considérer seulement deux classes d'expressions (les termes et les formules) nous allons envisager différents niveaux selon les règles de construction qui sont appliquées.

Le premier niveau étant celui des syntagmes élémentaires, les niveaux supérieurs s'obtiennent par utilisation des φ_p. On remarquera que, bien que dans la langue naturelle A il existe des mots qui jouent un rôle d'opérateur syntaxique (par exemple les prépositions, les conjonctions, etc.), le système fournit un syntagme au moins pour chaque mot et les fonctions syntaxiques sont considérées à part. Ce procédé a l'avantage de conserver la généralité de la construction; en contrepartie, il faudra distinguer, sur le plan de l'interprétation, les fonctions syntaxiques qui ont une fonction correspondante dans $\sum'$ (fonction sémantique) de celles qui ne sont pas susceptibles d'être interprétées.

Pour illustrer les constructions syntaxiques on pourra représenter $\varphi_p(s_i, s_j) = s_k$ par le schéma:

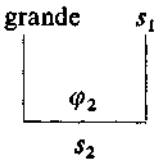


Exemple:



φ_1 associe la conjonction de coordination "et" à un adjectif. Le syntagme résultant s_1 se compose alors des éléments suivants:

$\langle K \rangle$: adjectif
$\langle vgp \rangle$: néant
$\langle vgc \rangle$: genre-nombre a la valeur féminin-singulier
$\langle cg \rangle$: ne gouverne rien
$\langle cd \rangle$: dépend d'un autre élément homogène
$\langle n \rangle$: n de niveau égal à 1



Le syntagme s_2 ainsi obtenu retrouve les codes identiques à celui d'un adjectif correspondant à un syntagme élémentaire; en outre il porte le numéro de niveau 2.

$$\text{A ce moment on a } s_2 = \varphi_2\ (\langle\text{grande}\rangle, s_1)$$
$$= \varphi_2[\langle\text{grande}\rangle, \varphi_1(\langle\text{et}\rangle, \langle\text{belle}\rangle)]$$

φ_2 indique une coordination complète, c'est-à-dire que s_2 comprend un opérateur "conjonction" ainsi que ses portées à droite et à gauche. Sur le plan de l'interprétation on fera correspondre à $\varphi_2(s_i, s_j)$ une fonction sémantique φ' et non à φ_1.

6.2. *Règles de constructions propres aux fonctions syntaxiques*

A chaque $\varphi_p(s_i, s_j)$ on associe un certain nombre de règles qui se répartissent en deux classes.

(*a*) *Règles de pertinence.* Il s'agit des règles qui precisent les valeurs prises par les différentes variables du code syntaxique appartenant aux syntagmes s_i et s_j. Ces règles permettent de définir la portée de l'opérateur syntaxique d'une part au moyen de la valeur prise par la catégorie syntaxique K, d'autre part au moyen de l'intervalle I

$$I = v_j - v_i - 1$$

qui permet ainsi le traitement de ce qu'on a l'habitude d'appeler "des constituants discontinus".

(*b*) *Règles de construction.* Ce sont celles qui définissent les valeurs prises par les différentes variables du syntagne s_k.

6.3. *Règles stratégiques*

Ce type de règle sort du cadre général des modèles que nous étudions. Elles concernent, non le passage progressif des termes aux groupes de termes plus complexes, mais l'ordre dans lequel on obtient ces groupes. A titre d'exemple citons les règles stratégiques suivantes:

Combinaison de gauche à droite par l'utilisation de deux listes (algorithme du centre de Milan).

Analyse prédictive (N.B.S. et Harvard).

Principe des priorités relatives (A. Auroux).

Principe de reconnaissance des groupes (M. Corbé, S. Lamb).

Principe du "nesting" (Georgetown, Ramo-Woldrige).

Principe des dépendances (Rand).

6.4. *La multiplicité des constructions syntaxiques*

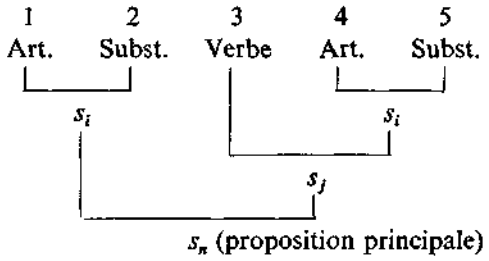
Etant donnée une suite de mots dans Λ, nous lui avons substitué une chaine de syntagmes. La multiplicité des constructions syntaxiques provient de causes différentes.

(*a*) A un mot de Λ correspondent plusieurs termes de $\mathscr{L}$ qui sont des homographes externes. Dans ce cas, ou bien la multiplicité disparait quand on atteint un certain niveau de complexité des syntagmes, ou bien (et cela est beaucoup plus rare) elle persiste jusqu'à l'obtention du syntagme s_n correspondant au niveau de la phrase.

Exemple du 1*er type:*

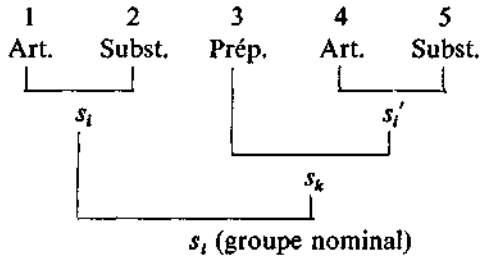
La	course	contre	la	montre	est	une	épreuve	difficile
1	2	3	4	5	6	7	8	9

La construction:



est incompatible avec la suite (6, 7, 8, 9).

Aussi, elle disparait et ne reste-t-il que la construction



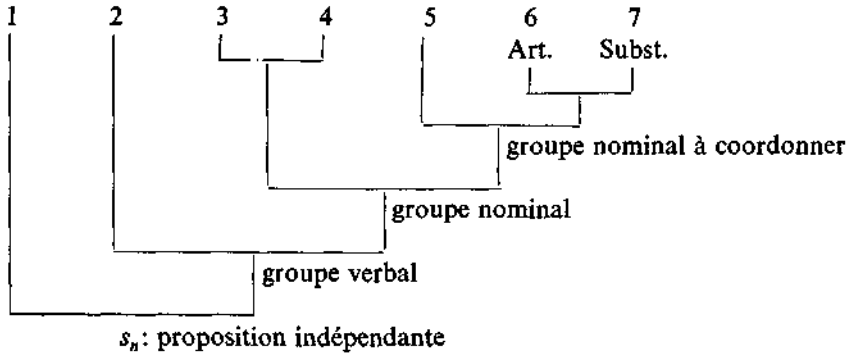
Exemple du 2ème type

Jean	prend	la	boule	et	la	lance
1	2	3	4	5	6	7

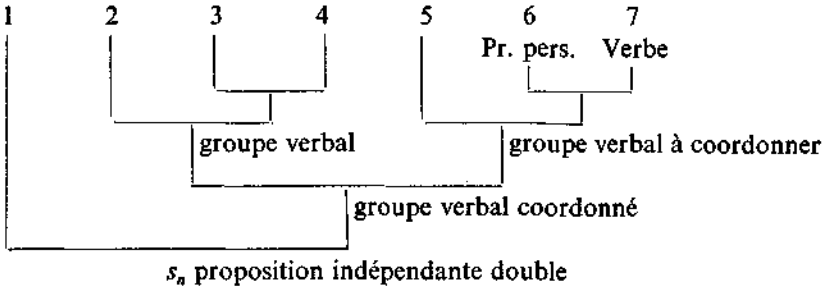
lance: $\begin{cases} s_7 : K \text{ substantif} \\ s_7' : K \text{ verbe personnel} \end{cases}$

la: $\begin{cases} s_6 : K \text{ article} \\ s_6' : \text{pronom personnel.} \end{cases}$

On obtient alors:



et aussi

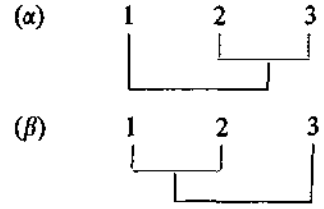


(*b*) La portée des fonctions syntaxiques est ambigüe

Exemple:

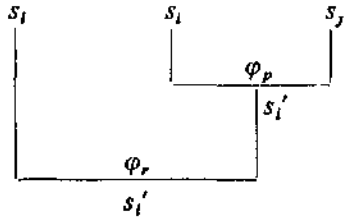
Subst/cas quelconque 1 — Subst/génitif 2 — Subst/génitif 3

fournit les deux constructions:

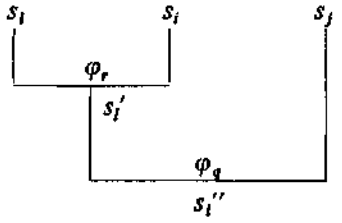


(*c*) Plusieurs fonctions syntaxiques sont autorisées par les règles de constructions:

Exemple: Soient s_j un groupe prépositionnel, $\varphi_p(s_i, s_j)$ où s_i est un groupe nominal, $\varphi_q(s_l, s_j)$ où s_l est un groupe verbal et $\varphi_r(s_l, s_i)$ associant un verbe à un groupe nominal complément. Alors, la chaine $s_l s_i s_j$ donne lieu aux constructions:



et



Dans le cas de multiplicité *b* et *c*, la réduction ne peut s'obtenir qu'au passage à l'interprétation, c'est-à-dire, au niveau des règles de construction sémantiques de $\sum$ [10].

7. POSITION DU PROBLEME DE LA MORPHOLOGIE

Si l'on se rapporte à la section 5.1 (choix des termes) nous considérons seulement le cas où le passage de Λ à $\mathscr{L}$ se fait au niveau du "*mot*" de la langue naturelle, l'élément commun entre Λ et $\mathscr{L}$ étant la "*forme*". Nous abordons l'analyse des tournures idiomatiques à la fin de cette leçon.

Ainsi, d'après nos définitions, l'analyse morphologique est constituée par le transfert

$$\langle \text{mot de } \Lambda \rangle :: = \langle \text{forme} \rangle \langle \text{information syntaxique} \rangle \langle \text{information sémantique} \rangle$$

à

$$\langle \text{terme ordonné} \rangle :: = \langle v \rangle \langle UL \rangle \langle K \rangle \langle Vgp \rangle \langle Vgc \rangle \langle Cg \rangle \langle Cd \rangle \langle \text{code sémantique} \rangle.$$

Le passage le plus simple est évidemment constitué par une table contenant toutes les formes de Λ, et en regard de chacune d'elles la liste de tous les termes qui lui correspondent.

Pour une langue aussi riche en formes infléchies que le russe ou le latin un tel procédé est peu économique. Il en va de même pour une langue dont la fabrication des mots composés est pratiquement illimitée, comme c'est le cas pour l'allemand. On peut donc imaginer que la "forme" soit considérée comme un élément complexe constitué d'une "*base*" (B) précédée d'un "*préfixe*" (P) et suivie d'un "*affixe*" (A).

$$\langle \text{forme} \rangle :: = \langle \text{préfixe} \rangle \langle \text{base} \rangle \langle \text{affixe} \rangle$$

Ainsi, au lieu de donner la liste des formes, on constitue un dictionnaire de bases ainsi qu'un dictionnaire de préfixes et d'affixes. On est alors conduit à envisager les décompositions $\langle P \rangle \langle B \rangle \langle A \rangle$ et à étudier les recompositions de manière à calculer un certain nombre des éléments des termes correspondants et à éliminer les compositions incohérentes.

Il est évident que plus la forme contient d'éléments simples et plus l'analyse linguistique et logique est compliquée, plus on réduit le volume du dictionnaire.

8. ELEMENTS DE L'ANALYSE MORPHOLOGIQUE

8.1. *Les données linguistiques—Application au russe*

Nous allons progresser par accroissements successifs de la complexité. Considérons tout d'abord une catégorie syntaxique K. A cette catégorie sont liées des variables grammaticales permanentes et contingentes (peut-être vides). Si plusieurs Vgc sont en cause on peut toujours se ramener à une seule dont le nombre de valeurs possibles est égal au produit du nombre de valeurs de chacune des précédentes.

Exemples: K: substantif. Vgc: *cas* 7 valeurs, nombre: 2 valeurs on obtient Vgc: *cas-nombre:* 14 valeurs.

Appelons $\alpha_1, \alpha_2, \ldots, \alpha_n$ les valeurs prises par cette Vgc liée à la catégorie syntaxique K.

Appelons $f_1, \ldots, f_n$ les formes d'un même mot (qui peuvent d'ailleurs ne pas être toutes différentes—homographies internes) correspondant aux valeurs $\alpha_1, \ldots, \alpha_n$ respectivement.

Enfin la connaissance de la langue nous montre qu'on peut trouver

$$\langle f_1 \rangle = \langle B \rangle \langle \delta_1 \rangle$$
$$\langle f_n \rangle = \langle B \rangle \langle \delta_n \rangle$$

où $\delta_i (i = 1, 2, \ldots, n)$ est une chaine de lettres variables appelée désinence et B une chaine de lettres fixes. Evidemment, on peut trouver des mots pour lesquels les formes $f_1, \ldots, f_n$ n'ont pas de base commune; on introduit alors plusieurs bases B, B', B'', etc.; en général si:

$$\langle f_i \rangle = \langle B \rangle \langle \delta_i \rangle, \text{ alors les concaténations}$$
$$\langle B' \rangle \langle \delta_i \rangle, \langle B'' \rangle \langle \delta_i \rangle \text{ n'existent pas.}$$

Ainsi, connaissant l'ensemble de couples $(\alpha_1, \delta_1, \ldots \alpha_n, \delta_n)$ appelé "*paradigme*" et la base B on peut construire et reconnaitre toutes les formes. Si, on se trouve devant un cas de bases multiples, on obtient le même résultat en se donnant les bases B, B', B'', etc., et les paradigmes π, π', π'' qui contiennent alors des couples α_i, δ_i vides si les bases sont mutuellement exclusives pour la forme f_i. Il ne faut pas confondre le couple α_i, δ_i vide (symbolisé par ϕ) qui interdit la construction de f_i par ce paradigme, de la désinence nulle —absence de lettres—(symbolisée par #).

Jusqu'ici nous nous sommes donc bornés à:

$$\langle P \rangle = \#$$
$$\langle A \rangle = \langle \delta \rangle$$

de sorte que:

$$\langle f \rangle = \langle B \rangle \langle \delta \rangle.$$

Les données de la langue russe (d'autres langues également) nous montrent que l'emploi d'un suffixe S permet de construire les formes appartenant au même mot, mais à une autre catégorie syntaxique K'.

Exemple: $S =$ *ЕЙШ* sert à la formation des superlatifs à partir d'une base d'adjectif.

D'autres suffixes permettront de passer d'une base de verbe à la formation des participes, etc.

A ce niveau $\langle A \rangle = \langle S \rangle \langle \delta \rangle$.

Si, pour certaines catégories syntaxiques le mot russe prend la forme réfléchie terminée par *Ся* ou *Сь* on voit qu'en définitive A est de la forme:

$$\langle A \rangle = \langle S \rangle \langle \delta \rangle \langle S' \rangle$$

en appelant S' ce deuxième type de suffixe (réflexivité).

Arrivés à ce point, on peut introduire d'autres suffixes en généralisant le processus de passage d'une catégorie syntaxique à une autre. Nous appellerons "dérivation" ce processus généralisé, que ce dernier s'obtienne au moyen d'un suffixe ou non.

Exemple: En français la base UTIL donne les formes de l'adjectif UTILE et UTILES; le suffixe ITE fournit les formes du substantif UTILITE, UTILITES; le suffixe MENT concaténé à la forme de l'adjectif féminin singulier conduit à la forme de l'adverbe UTILEMENT.

En russe, le suffixe OCT (ou ECT) peut servir à la formation d'un substantif à partir d'un adjectif. Par contre la forme courte neutre singulier peut servir (donc sans emploi de suffixe) à exprimer la forme de l'adverbe dérivé d'un adjectif. Il en va d'une manière analogue pour les adjectifs dont certaines formes ont aussi ou bien uniquement valeur de substantif.

La dérivation introduite jusqu'ici par adjonction de suffixes peut être étendue au cas de l'utilisation de préfixe.

Exemple: en russe *НЕ* préfixe de négation

по préfixe d'atténuation du comparatif, etc.

Nous sommes ainsi arrivés à considérer la forme f comme élément complexe composé de

$$\langle f \rangle :: = \langle P \rangle \langle B \rangle \langle S \rangle \langle \delta \rangle \langle S' \rangle.$$

8.2. *Les problèmes de formalisation*

Les considérations qui précèdent permettent d'éclairer et de préciser certaines notions indiquées dans la leçon précédente, comme par exemple, la notion d'unité lexicale.

Alors que le concept de "mot" dans la langue naturelle n'est pas clairement défini, celui d'"unité lexicale" peut maintenant être parfaitement précisé. En effet, en français, est-ce que le SON (du tambour) est un mot différent du SON (du blé)?

Est-ce que le VEAU correspond à deux mots puisque sa traduction en anglais sera différente suivant qu'il est considéré comme animal vivant ou comme viande? est-ce que UTILE, UTILITE, UTILEMENT, INUTILE, etc., sont des mots différents? Dans l'affirmative, est-ce qu'alors on considère comme deux mots différents les formes d'un verbe conjugué d'une part et les formes de son participe passé à valeur d'adjectif d'autre part?

Sur le plan du système formel nous disposons de bases, de préfixes, et d'affixes (ces derniers décomposables en trois éléments).

A une catégorie syntaxique nous avons associé une *vgc* (éventuellement généralisée) représentée par l'ensemble $(\alpha_1, \ldots, \alpha_n)$ de ses valeurs possibles. Les préfixes et suffixes de dérivation introduits dans le système (ainsi que les dérivations obtenues sans modifications de la forme) permettent de construire des réseaux de connections entre différentes catégories syntaxiques. Un tel réseau sera appelé "classe morphologique" [11]. Il existe alors une base (au moins) telle que les formes construites au moyen des règles de ce réseau puissent respectivement correspondre à toutes les catégories syntaxiques qui y sont réunies et aux différentes valeurs de la *vgp* de chacune de ces dernières. L'ensemble des formes générées ou reconnues à partir de cette base appartenant à une certaine classe morphologique est appelé "unité lexicale"—(*U.L.*). Le numéro d'*UL* est alors indiqué sur cette seule base.

Il peut arriver maintenant deux cas:

(*a*) La base n'est pas susceptible de générer toutes les formes parce que l'unité lexicale correspondante n'admet pas certaines dérivations ou n'admet pas toutes les valeurs possibles d'une *vgp*.

On est alors en présence d'une *UL* comportant des défectivités.

(*b*) Toutes les formes peuvent être générées, mais une seule base ne suffit plus. On introduit alors le nombre de bases nécessaires à la génération ou à la reconnaissance de toutes les formes qui existent dans cette classe. En général, nous l'avons remarqué en section 8.1, les bases s'excluent mutuellement pour la construction d'une forme particulière. Cependant il peut arriver que deux ou plusieurs bases subsistent:

Exemple: En français PAYER

Bases: PAY et PAI

Indicatif présent—1ère personne du singulier PAY-E ou bien PAI-E. Ce cas particulier n'introduit aucune difficulté de principe. Les différentes bases qui concourent ainsi à la construction ou à la reconnaissance de toutes les formes possibles dans une classe morphologiques recevront le même numéro d'unité lexicale.

Ainsi PAY et PAI ont le même numéro d'unité lexicale.

La base UTIL qui donne naissance aux formes

$$\left.\begin{matrix}\#\\ \text{IN}\end{matrix}\right\}\ \text{UTIL}\ \begin{cases}\text{E}\\ \text{ES}\\ \text{ITE}\\ \text{EMENT}\end{cases}$$

correspond à une seule unité lexicale.

Ainsi *l'étentue de la notion d'unité lexicale est liée à la structure de classe morphologique* et l'on voit maintenant pourquoi il n'était pas possible d'en donner plus tôt une définition plus précise.

Les réseaux adoptés pour la classification morphologique au C.E.T.A.-G. sont donnés par la référence [12].

8.3. *Le code morphologique*

Si l'on se rappelle que nous avons défini l'analyse morphologique comme l'opération qui permet de passer de la forme (suite de lettres d'un mot de la langue naturelle A) au terme (suite de codes dans le langage artificiel $\mathscr{L}$) et que nous ne considérons plus un dictionnaire de formes mais un dictionnaire de bases et un dictionnaire de préfixes et d'affixes, nous nous rendons compte qu'un code supplémentaire doit accompagner les bases, affixes, et préfixes pour obtenir la reconnaissance ou la construction des formes; c'est ce code que nous appelons "*code morphologique*". Des études générales et des descriptions détaillées de ce code pour le russe sont données aux références [8], [11], [12], [13]. Nous nous bornerons, ici, à montrer comment interviennent d'une manière générale les divers éléments du code morphologique.

Etant donné que dans tous les cas nous devons fabriquer le ou les termes ordonnés correspondant à une forme c'est à dire déterminer:

$\langle v \rangle$: numéro séquentiel
$\langle UL \rangle$: numéro d'unité lexicale
$\langle K \rangle$: catégorie syntaxique
$\langle Vgp \rangle$: valeur de chacune des variables grammaticales permanentes
$\langle Vgc \rangle$: valeur de la variable grammaticale contingente
$\langle cg \rangle$: code de gouvernement
$\langle cd \rangle$: code de dépendance
$\langle c \rangle$: code sémantique.

Nous allons montrer progressivement ce qui est simplement recopié et ce qui est calculé lorsque l'analyse morphologique devient de plus en plus complexe.

Tout d'abord on suppose que $\langle f \rangle :: = \langle B \rangle \langle \delta \rangle$, que les formes d'une même unité lexicale appartiennent à la même catégorie syntaxique. Alors seules les valeurs de la *Vgc* correspondant à la valeur de δ sont les quantités à calculer. Eventuellement, les valeurs des *Vgp* le seraient aussi si ces dernieres, dans certains cas exceptionnels dépendent de la valeur de *Vgc*. On dispose alors de paradigmes dans cette catégorie et la connaissance du numéro de paradigme permet de résoudre le problème. Pour éviter un nombre trop grand de paradigmes on peut les grouper en "*systèmes de désinences*". A l'intèrieur d'un tel système le paradigme correct est retrouvé au moyen d'un critère. Ainsi le code morphologique lié à une base devra comprendre tous les codes du terme sauf $\langle Vgp \rangle$ qui est remplacé par un numéro de système de désinences ou de paradigme; le code morphologique lié à une désinence devra indiquer dans quelles catégories K cette dernière intervient et la table $\{\delta_i, \alpha_i\}$.

Si maintenant, on se donne les classes morphologiques comme elles ont été définies plus haut, le code morphologique se présente de la manière suivante:

Dans le réseau de connections entre catégories syntaxiques ou encore à l'intèrieur d'une même catégorie au moyen d'un élément de dérivation (tel que HE en russe, IN en français, UN en allemand, etc.) certains de ces aiguillages sont valables pour toutes les unités lexicales

par exemple, le préfixe НАи pour l'adjectif russe; ces auguillages systématiques sont représentés par des "boîtes" en pointillé sur les réseaux des documents cités; aussi n'apparaissent-ils pas dans le code morphologique. Par contre les autres possibilités de dérivation dépendent de l'unité lexicale; aussi la base ou les bases représentatives de cette unité lexicale doivent-elles contenir dans leur code morphologique les indications relatives à ces dérivations.

Dans ce cas le code morphologique contient le numéro de classe morphologique, le numéro d'unité lexicale, les valeurs des *Vgp* (éventuellement en fonction de *Vgc*), les possibilité de dérivation et les défectivités particulières. L'analyse morphologique calcule alors *K*, *Vgc*, éventuellement *Vgp*, *cg*, *cd*, et copie le code sémantique.

v est évidemment donné par le texte lui-même. En outre, pour chaque terme, ou plus précisément pour chaque famille d'homographes internes, l'analyse morphologique note le chemin parcouru dans le réseau; nous appelons "code dérivation" ($C\Delta$) cette indication qui sera reprise plus en détail au cours de la leçon suivante.

Ainsi, le terme ordonné se présente maintenant sous la forme:

$$\langle \text{terme ordonné} \rangle :: = \langle v \rangle \langle UL \rangle \langle K \rangle \langle C\Delta \rangle \langle Vgp \rangle \langle Vgc \rangle \langle cg \rangle \langle cd \rangle \langle c\sigma \rangle.$$

Enfin, pour traiter de manière commode les irrégularités morphologiques très exceptionnelles, on ajoute au code morphologique un code spécial dit "code exception" dont la valeur indique si l'analyse morphologique a lieu normalement ou bien si l'on agit comme dans le cas d'un simple dictionnaire de formes où toutes les informations n'ont qu'à être recopiées.

9. REALISATION DE L'ANALYSE MORPHOLOGIQUE

La réalisation de l'analyse morphologique est liée à la consultation du dictionnaire ou plus exactement des dictionnaires (dictionnaire de bases, dictionnaire de préfixes et d'affixes).

Les études de D. Hays, celles de S. Lamb poursuivies par celles de G. Veillon ont permis l'acquisition des résultats suivants:

L'idée selon laquelle la traduction automatique consisterait principalement à faire des consultations de dictionnaires et de tables de taille extraordinaire nécessitant des mémoires de capacité énorme et pour lesquelles il faudrait trouver un temps d'accès raisonnable, est sèrieusement attaquée: en effet, pour une langue comme le russe un corpus très vaste peut être traité avec un nombre de mots relativement restreint; si l'on considère la notion d'unité lexicale avec tout le potentiel de dérivation qu'elle représente, un petit nombre de bases suffit pour réaliser l'analyse morphologique d'un grand nombre de textes. Il en résulte que si les dictionnaires peuvent être consultés en mémoire rapide, simultanément, la complexité de l'analyse morphologique apporte ses fruits. La capacité en mémoire rapide des calculateurs électroniques de grand puissance est actuellement telle, que des procédés d'organisation et de programmation efficaces permettent le stockage de 20.000 bases russes ainsi que de quelques centaines d'affixes.

Le détail des opérations sort du cadre que nous nous sommes proposé; nous mentionnerons seulement la suite des phases de la réalisation de cette analyse morphologique.

(*a*) Entrée dictionnaires en mémoire rapide. Les dictionnaires ne comprennent que les "en-tête" d'articles mais tous les articles sont représentés.

(*b*) Défilé du texte et consultation des dictionnaires pour trouver les décompositions possibles de la forme. On en déduit une chaîne d'addresses correspondant aux bases affixes et p xes découverts.

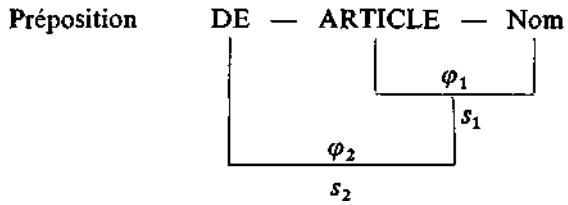
(*c*) Entrée des codes morphologiques des éléments précédents à la place qu'ils occupaient.

(*d*) Calcul des éléments du code syntaxique.

10. LA SEPARATION DES NIVEAUX MORPHOLOGIQUES— SYNTAXIQUES ET SEMANTIQUES

Cette question est extrêmement importante pour la réalisation des modèles et surtout pour le transfert impliqué par la traduction automatique. Elle trouve sa place ici car la dérivation telle que nous l'avons décrite peut avoir une influence directe sur chacun de ces niveaux. En premier lieu, le choix du terme dans $\mathscr{L}$ est une option fondamentale. Nous avons montré en quoi consistait ce choix dans la deuxième leçon. Etant donné que nous considérons toujours la "*forme*" de tout mot de A pour générer les termes de $\mathscr{L}$ (termes qui pour une forme donnée sont mutuellement exclusifs), que dans tout terme se trouve un syntagme élémentaire, le transfert de $\mathscr{L}$ à $\mathscr{L}'$ peut présenter des différences de niveaux.

Par exemple, soit un substantif au génitif en russe qui donne dans le modèle un seul terme, donc un seul syntagme élémentaire. Dans le modèle français on peut trouver en correspondance la chaine présentant la structure:



Ainsi l'équivalent dans le modèle français du syntagme élémentaire dans le modèle russe se trouve être un syntagme complexe de niveau 2. On aboutira alors à une traduction française de trois mots. Il en résulte que chaque modèle doit posséder son ensemble de fonctions.

Si nous considérons toujours deux types d'expressions dans $\mathscr{L}$: les termes τ et les formules $\mathscr{F}$, le niveau morphologique consiste à segmenter le sous-ensemble τ, le niveau syntaxique à segmenter le niveau $\mathscr{F}$.

Les éléments du langage au niveau $\mathscr{F}$ sont:

—des variables: catégories syntaxiques, variables grammaticales, codes de dépendance et de gouvernement, numéro séquentiel . . .

—des opérateurs: les fonctions syntaxiques φ
auxquels on ajoute une liste de règles de construction.

Le niveau sémantique est constitué par un autre langage artificiel $\sum$ [10] dans lequel interviennent:

—des variables: champs sémantiques, variables sémantiques, unités sémantiques . . .

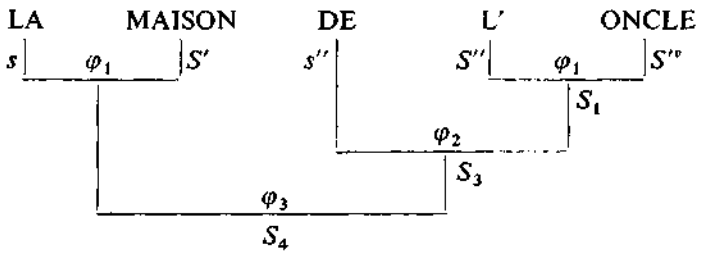
—des opérateurs: les fonctions sémantiques φ'
auxquels on ajoute des règles de construction.

Ainsi le code sémantique ($c\sigma$) que nous avons mentionné jusqu'ici sans le faire intervenir contient-il pour chaque unité lexicale la liste des variables ainsi que leurs valeurs qui doivent

lui être associées. Une formule de $\mathscr{F}$ sera dite "valable" au sens de "l'interprétation" que nous avons défini s'il existe une structure de fonctions φ' déduites des fonctions φ qui y figurent, obéissant aux règles de construction de $\sum$.

Ainsi, alors que les fonctions φ portent sur deux syntagmes (et ceci, surtout pour des raisons de commodité), les fonctions φ' portent sur un syntagme (toujours complexe) et peuvent mettre en jeu autant de variables qu'il est nécessaire.

Exemple:



$$S_4 = \varphi_3(S_2, S_3) = \varphi_3\{\varphi_1(S, S'),\ \varphi_2[s'', \varphi_1(S''', S'')]\}.$$

Alors que les constructions s_1, s_2, s_3 n'ont pas de formules interprétatives dans $\sum$ (mais seulement des termes), s_4 en possède une.

En effet, on fera correspondre à φ_3 une fonction sémantique φ_1' qui indique la relation d'appartenance. On pourra alors remarquer que cette relation interprète dans le modèle russe, une fonction syntaxique qui porte sur un adjectif d'appartenance et un nom.

Si l'on possède la liste des φ' du langage $\sum$ on voit qu'à certaines fonctions syntaxiques de $\mathscr{L}$ correspondent une ou plusieurs fonctions de cette liste.

Ainsi, dans l'exemple précédent, à φ_3 correspond aussi la fonction sémantique φ_2' indiquant "l'activité-objet" comme ce serait le cas pour: LA CULTURE DU RIZ où l'on obtient le même φ_3. La donnée d'une part de φ_3 et d'autre part des codes sémantiques contenus dans les syntagmes élémentaires mis en cause doivent permettre le choix de l'interprétation entre φ_1' et φ_2' par application des règles de construction de $\sum$. Sans détailler davantage la structure de $\sum$, nous avons indiqué ce qui servait à comprendre d'une part le transfert par le chemin IV (figure 3—Première leçon) et d'autre part l'influence de la dérivation.

11. DERIVATION ET MORPHOLOGIE

La dérivation a permis la constitution des classes morphologiques et le potentiel d'un grand nombre de formes aux unités lexicales. Sur le plan purement morphologique la dérivation qui agit aussi sur le plan syntaxique (en russe: suffixe de participe) ou sur le plan sémantique (en russe: préfixe HE) ne se distingue pas de celle, extrêmement particulière et simple, que l'on fait implicitement au moyen de désinences et de paradigmes.

Cependant, si l'on aborde par ce procédé la reconnaissance des mots composés (comme il peut être utile de le faire en allemand) il faut alors considérer que les bases elles-mêmes (ou peut-être seulement certaines d'entre elles) deviennent des affixes possibles.

Ce point modifie légèrement le programme de consultation des dictionnaires mais ne met pas en cause le principe général de l'analyse morphologique.

En résumé, l'avantage majeur, de la dérivation en morphologie revet un aspect d'efficacité économique car elle réduit considérablement le volume du dictionnaire.

12. DERIVATION ET SYNTAXE

Nous avons mentionné l'existence de catégories syntaxiques *K*. Au niveau des syntagmes élémentaires (ceux qui sont déduits directement des formes) on obtient une liste de ces catégories. La liste s'accroît lorsque l'on passe aux syntagmes complexes par utilisation des fonctions syntaxiques. La dérivation a une influence directe sur la constitution de cette première liste. Cette influence provient du traitement des mots composés.

Nous empruntons les exemples qui suivent à l'allemand qui est la langue naturelle qui se prête le mieux à ce genre de problème.

Exemples:

Informationstheorie
Vaterhaus
Kartoffelsuppe
Regenschirm

son des formes qui contiennent une fonction syntaxique entre deux substantifs.

Avec le groupement Adjectif-Adjectif: taubstumm
Adjectif–substantif: Unterarm
Adjectif–verbe losreissen
Substantif–verbe: radfahren.

On peut même aller plus loin et fabriquer ainsi, un syntagme à valeur de substantif à partir d'un substantif et d'un autre substantif lui-même dérivé d'un verbe.

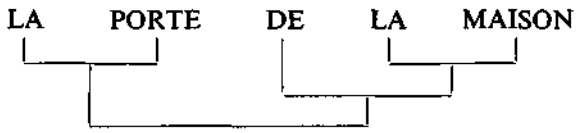
Exemple:

Radfahrer.

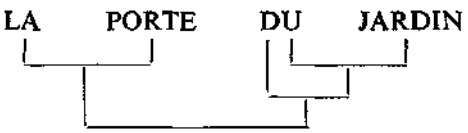
A une forme telle que Vaterhaus correspond toujours un syntagme élémentaire, mais qui sera équivalent à un syntagme s_3 donné dans l'exemple de la section 10.

Il peut arriver par contre que la forme donne naissance à deux syntagmes élémentaires de numéros séquentiels consécutifs lorsque la relation syntaxique n'existe pas:

Exemple: en français



et



La liaison DE LE contenue dans la forme DU ne se trouvant pas parmi les règles de construction des formules de modèles français, la contraction est régénérée en deux éléments distincts.

13. DERIVATION ET SEMANTIQUE

L'influence de la dérivation peut être très importante sur le plan de l'interprétation du système $\mathscr{L}$ c'est-à-dire dans le langage $\sum$.

Tout d'abord, on peut considérer un certain nombre de cas simples:

Exemples: En russe le préfixe $\pi 0$ devant un comparatif, indiquant l'atténuation.

Le préfixe de négation: en russe HE en allemand UN, etc.

Ensuite, la dérivation peut conduire, par le fait qu'elle donne déjà des structures syntaxiques complexes, à la recherche de fonctions sémantiques avant celle de structures syntaxiques plus compliquées. C'est le cas pour les exemples:

INFORMATIONSTHEORIE	relation sémantique	φ' déterminative
VATERHAUS	relation sémantique	φ' d'appartenance

Enfin elle peut intervenir au niveau sémantique pour l'élimination des découpages non valables des mots composés.

Exemples:

	KLAMMERSATZ
fournissant	KLAMMER/SATZ
et	KLAMM/ERSATZ

14. TRANSFERT DE LA DERIVATION

Que la dérivation intervienne sur l'un ou plusieurs des trois niveaux envisages (morphologie, syntaxe, sémantique), elle porte toujours sur les unités lexicales appartenant à une certaine classe morphologique. Au cours de l'analyse morphologique, le "chemin" suivi dans le réseau correspondant à la classe morphologique indexée à la base déduite de la forme à étudier, est noté sous la forme du code de dérivation $C\Delta$ dans le terme ordonné.

Par ce procédé, nous avons donc:

Au départ une forme f d'un mot de la langue naturelle-source A. Le découpage de f en *PBA* (préfixe, base, affixe) et l'analyse morphologique permettant de trouver les éléments du terme correspondant dans $\mathscr{L}$. Parmi les éléments de ce terme nous nous interessons aux suivants:

—la catégorie syntaxique K

—le code de dérivation $C\Delta$

—certains éléments du code sémantique: unité sémantique $U\sigma$ et champ sémantique γ.

Le couple $(U\sigma\gamma)$ fait correspondre une unité lexicale UL' de $\mathscr{L}'$, modèle de la langue naturelle cible.

D'autre part, la structure de φ' de la phrase à laquelle appartient f fait correspondre une structure syntaxique de φ de sorte que dans le syntagme élémentaire appartenant à $\mathscr{L}'$, la catégorie syntaxique est devenue K'.

Alors plusieurs cas se présentent:

(a) 1. Dans la classe morphologique à laquelle appartient UL' il existe une dérivation analogue à $C\Delta$ conduisant à la catégorie K'.

2. La construction de cette dérivation est possible au moyen des règles générales relatives à $C\Delta$.

Exemples: La forme russe ТРУДНО provenant de la base ТРУДН qui appartient à la classe morphologique "adjectif" conduit à:

K = adverbe

$C\Delta$ = Adjectif forme courte neutre → adverbe: posons $C\Delta = 1$

$us, \gamma = us_1, \gamma_1$

us_1, γ_1 conduit au numéro d'unité lexicale UL' représentée dans le dictionnaire par la base DIFFICILE indexée dans la classe morphologique (du modèle français) "adjectif".

Le transfert syntaxique à donné $K' = K$: adverbe. Enfin, la dérivation analogue existe en français, et pour l'adjectif "DIFFICILE" elle se fait au moyen de la règle générale:

⟨forme du féminin singulier⟩⟨MENT⟩.

On obtient alors DIFFICILEMENT. Il n'y a pas de problème.

(*b*) La condition 1 est toujours satisfaite mais la condition 2 ne l'est plus.

On peut alors reprendre l'exemple de la dérivation précédente en changeant d'unité lexicale.

Soit

МИЛО → GENTIMENT.

Or cette forme française n'obéit pas à la règle précédente; elle se trouve donc ailleurs dans le dictionnaire. En fait la base GENTIL indique que la dérivation $C\Delta = 1$ ne peut être calculée et qu'il faut la chercher à une certaine addresse.

(*c*) La condition 1 n'est pas vérifiée. On se trouve dans un cas analogue au précédent, mais cette fois il y a changement d'unité lexicale.

Le traitement de ces deux derniers cas implique la reconnaissance des bases qu'il convient de choisir dans le dictionnaire de français. On trouvera les détails de la solution de ce problème à la référence [14]. En bref, lorsqu'on fait la sélection des bases utiles avant la consultation du dictionnaire de langue cible, on étudie la possibilité effective de dérivations et les codes associés au numéro d'Unité lexicale permettent de calculer l'adresse de la base correspondante.

BIBLIOGRAPHIQUES

[1] JEAN LADRIERE: Les Limitations Internes des Formalismes (1957).
[2] S. C. KLEENE: Introduction to Metamathematics (1952).
[3] A. CHURCH: Introduction to Symbolic Logic.
[4] SYDNEY LAMB, A. HUDSON et C. JOHNSON: A System for Analysing Russian Texts (Internal memorandum) (1960).
[5] F. GENUYS: Conférence à l'AFCALTI—Sémissaire sur les Langages de Programmation (1962).
[6] J. W. BACKUS *et al.*: Report on the Algorithmic Language. ALGOL 60. *Numerische Mathematik* (1960), 2, 106–136.
[7] B. VAUQUOIS: Congrès AFCALTI (1960).
[8] G. VEILLON: Thèse de 3e Cycle (Juin 1962).
[9] A. AUROUX: Thèse de 3e Cycle (Juin 1962).
[10] D. AUGEREAU: Thèse de 3e Cycle (en cours).
[11] B. VAUQUOIS et J. VEYRUNES: Document G–100–C du CETA–G.
[12] D. ANDRON, L. TORRE, A. GAGNY et H. DE CROUSNILHON: Documents G–101–4, G–102–3, G–103–2, G–104–1, du CETA–G.
[13] J. VEYRUNES: Congrès AFCALTI (1960).
[14] M. BERTAUD: Thèse de 3e Cycle (en cours).