It is now quite generally supposed that a natural language cannot be adequately described by a CF grammar. This opinion was first advanced by Chomsky who discussed this problem from the point of view of phrase structure grammars. He presents there a fragment of a CF phrase structure grammar in terms of noun phrases NP, verb phrases VP, etc., which are familiar from immediate constituent analysis. These rules cannot treat verbal selection rules properly; Chomsky (ch. 8) had already tried himself to correct this defect within the framework of a CF phrase structure grammar, but the difficulties he encountered seem to have persuaded him that only a transformational grammar could handle such a problem.

Harman proposed another solution to the problem of treating verbal selection rules in a CF grammar; he added a set of subscripts to the CF rules used in Chomsky, which were chosen so that only those subjects and objects which satisfied the selection rules could appear with a given type of verb. Chomsky showed that this method would not suffice if the sentences subscripted as Harman had suggested were themselves embedded in complement constructions. Thus, where Harman's system will not generate such aberrant sentences as Bill elapsed, it will not be able to exclude the generation of such a sequence when it is embedded in a complement construction, as in John persuaded Bill to elapsed.

Further arguments for the inadequacy of a CF grammar were adduced from the fact that sentences containing respectively cannot be assigned an appropriate structure in the framework of a CF grammar. This was noted by Chomsky in his discussion of the algebraic language w w; the relation between this language and sentences containing respectively was discussed by Hillel & Shamir, and then taken up again by Chomsky together with examples taken from the comparative construction in English. Later, Postal exhibited a construction in Mohawk which is similar to the one with respectively, and like the latter, is recursively extendable to sentences of theoretically unbounded length.

As a result of these considerations, Chomsky concluded that a coherent description of recursively embedded sentences or of verbal selection rules could not be obtained in a natural way by any CF grammar, and that consequently no CF grammar could adequately describe a natural language. However, it turns out that this question is not so easily disposed of as it would appear, and recent work by Joshi & Levy shows that a CS grammar containing rich context-dependent rules can be used to analyze trees that describe a CF language. They did this by an extension of a theorem of Peters & Ritchie, who showed that CS rules of a certain type can be used not to generate sentences, i.e., not to characterize them, but only to verify their well-formedness, by applying the context-dependent parts of these rules as constraints on the set of trees that schematize these sentences. In this case, the language described by these trees is a CF language.

Joshi & Levy generalized the kinds of CS rules that can be used for this result and defined CS rules that can describe conditions on the context whose action is close to that of certain transformations. These rules are expressed as Boolean combinations of predicates that describe the left and/or right context of a node, or the upper and/or lower contexts (the nodes above and below a given node). Roughly speaking, a tree is said to be analyzable with respect to a grammar containing such rules if one of the rules is satisfied at each node of the tree. In that case, the language which consists of the terminal strings of all the trees analyzed by the grammar is a CF language, even though the rules take the context into account. Hence these terminal strings can be described by

\[ \text{Note that the formalism used by Joshi & Levy for displaying conditions on trees is close to the notation used for rewrite rules, and can lead to some confusion. It need only be remembered that these context-dependent rules are not used to generate structures.} \]
some CF language.

Now the string grammar proposed by Harris and which analyzes English (Sager) and French (Salkoff) can be shown to be of just the form described by Joshi & Levy. It contains CS rules of the type described by them, and is used to analyze a tree, rather than to generate it. It would thus appear that English or French can be described by some CF language, although the string grammar gives no clear clue as to what its form would be. I shall show here that such a CF grammar can be written for French, and that it can treat, in a linguistically appropriate fashion, the problem of the expression of verbal selection rules in nested complement constructions. I have chosen French because systematic data giving a wide coverage of the French lexicicon are available (Gross, Boons et al.); however, the very nature of this construction makes quite plausible its extension to other natural languages. Only the method used will be outlined in this brief article, and an example of its application to embedded complement constructions; for more details, consult Salkoff (chap. 3).

I now construct CF rules that correctly describe sentences in which related pairs like verb-subject, verb-object, etc., that are linked by a relation of selection, may be separated by constructions of unbounded length. Each such CF rule is the expansion of a sentence schema $S$. The verbal selection rules are accounted for in this method by separating the semantic function of a selection rule, namely, the exclusion of certain noun sub-classes, from the syntactic relation between the pairs carrying this function (generally, a verb and a noun phrase). Each selection rule is decomposed into two independent parts: one part is the choice of a noun not classified in certain noun sub-classes, in such a way as to express the semantics of that selection rule; the second part is the use of the noun phrase containing this N for the subject or object of a given verb in a rule schema, which amounts to satisfying the complete verbal selection rule.

Conjunctive sequences, including sentences containing respectivement can be handled by this method, but not within the strict mathematical framework of a CF language. The resulting CF grammar of French can be compared with a transformational grammar. The complex symbols are useful in order to explain clearly the process of sentence embedding; they will be eliminated in a second step and replaced by the noun phrases without subscripts used in the verbal selection rules.

Main rule schema.

For clarity, I shall use only the subscript $F$ (s, o, or io) in the rules for $S$. Only an abbreviated list of these rules can be given here; for a complete list, cf. Salkoff. A first subgroup of rules contains non-sentential objects:

1. The base rules

In order to set forth the selection rules as clearly as possible, I shall begin by using in the rules developing $S$, noun phrases bearing three subscripts, i.e., complex symbols:

$$ S \rightarrow NP_s \times V $$

where $x$ is a function $F$: subject $s$, object $o$, or indirect object $i o$; $y$ is the morphology $M$: singular, plural,...; and $z$ is a semantic sub-class $S$: these

sets have no elements in common.

With this notation, typical rules for $S$ will have the following form:

$$ S \rightarrow NP_{s,y,z} \times V_{1, o,y,z} $$

$$ S \rightarrow NP_{s,y,z} \times V_{2, P, NP_{o,y,z}} $$

$$ S \rightarrow NP_{s,y,z} \times V_{3, P, NP_{o,y,z}, P, NP_{10,y,z}} $$

The verb is subscripted according to the complements it takes.

In this notation, the CF rules no longer constitute a strict constituent grammar of the type discussed by Chomsky. My notation brings out the grammatical relations between the elements of the sentence schemata, which is not possible in a direct way in a phrase structure grammar. The complex symbols are useful in order to explain clearly the process of sentence embedding; they will be eliminated in a second step and replaced by the noun phrases without subscripts used in the verbal selection rules.

The new notation $NP_s$ denotes a noun phrase having a double function $F$: it must be an accept-
able object of the verb V30 which precedes §, and also an acceptable subject of the main verb of S1. The sentence schema for S1 is a sentence deformation (in Harris' terminology); there are about ten such deformations in French. Another one is the following:

(5) S → NP₈ t V32 S₃ (Max apprend ...) S₃ → ɂ₁₀ NP₈ V₁ ... Paul at dormir) ...

Each such schema Sᵢ contains as many rules as S itself.

With the schemata Sᵢ, I can account for the recursive embedding of sentences, like Luc convainc Paul d'apprendre Max ... Paul lit) ...

(6) a NP₁ → NP₈ que NP₈ t V30 S₁ pro (l'homme que Max convainc)
   b S₁ pro → (Φ) de V₁ ... de dormir(
   c (Φ₈ de V₂ NP₈ ... d'ôter cela) ...

Here, the symbol Φ₈ is a dummy element standing for the noun phrase, like the same subscripts, at the head of NP₁. It is marked by the same selectional features as ΦNP₈ and will be used to 'transmit' this selection through embedded sentences. Such dummy elements come close to certain pronouns found in relative clauses without antecedent, like ce in: d'ai acheté ce que Max a sculpté.

A second type of relative clause is this:

(7) NP₂ → NP₈ que S₈ pro (le livre que Paul lit)
   S₈ pro → NP₈ t V₂ (Φ) ... Paul fait de ...
   NP₈ t V₃₀ S₃₁ pro ... Max convainc...
   S₃₁ → NP₈ de V₂ (Φ) ... Paul de faire de ces articles)
   NP₈ de V₄ (Φ) ... Paul de faire de ces articles)
   ...
   NP₈ t V₃₂ S₃₁ pro ... Max apprend...
   S₃₁ → ɂ₉₈ NP₈ d V₂ (Φ) ... Paul ...

With these rules, it is possible to describe recursively embedded sentences inside relative clauses, although the complex symbols give us no indication yet as to how the verbal selection rules are to be satisfied.

§2. Selection Rules

According to the kind of noun allowed as subject, or as direct or indirect object, a verb is said to select for that sub-class. The majority of the selection rules thus concern the following three rules for S:

(8) a S → NP₈ t V₃ NP₉ (Luc porte un chapeau)
    b S → NP₈ t V₃ P₀ NP₉ (Max dépend de Luc)
    c S → NP₈ t V₄ P₀ NP₉ (Max attribue la médaille à Luc)

The selection rules vary with the preposition P_i for verbs V₃ and V₄. In the sentence analyzer based on the string grammar, these selection rules are contained in a system of contextual rules attached to each lexical entry for a verb that can appear in (8). Experience shows that five noun sub-classes are needed for such a system of selection rules: N₁, 'time'; Nₛ, sentential; N₉₉, 'human'; N₉₉, concrete; and N₉₉, nominalizations. These sub-classes are used in the verbal entries to indicate the unacceptable contexts for a verb classified in V₃, V₄, or V₉. The analyzer then uses these contextual rules to disallow an unacceptable decomposition in a sentence analysis.

These contextual rules can be replaced by CF rules in the following way. Let

(9) N \{N₁ + N₉ + ...\} = N'

denote any noun except one belonging to sub-class N₁ or to N₉, etc.; the bar \ means 'minus'. If N' is substituted for the noun N in any NP, and carried over into every rule developing NP, the terminal rule for the noun in NP will be

(10) N' → N₉₉, where N₉₉ is a lexical entry.

Each choice for N₉₉ is compared with the list of sub-classes N₁, N₂, ..., attached to N. If N₉₉ belongs to any of these sub-classes, it is discarded; if N₉₉ doesn't belong to these sub-classes, the conditions expressed in (9) are satisfied. Now, if the selection rule of a given verb is that sub-classes N₁, N₂, ..., are unacceptable as subject (object), then the noun phrase containing N₉₉ satisfies that selection rule, and will be the only noun phrase permitted in that syntactic position.

I now define noun phrases GN containing all the combinations of excluded noun classes from the five named above (there are 3! such GN):

(11) a GN → N, if no sub-classes are excluded;
    b GN₁ → N\{N₂\}; GN₂ → N\{N₉\}; ...
    GN₉ → N\{N₉₉\};
    c GN₁₉ → N\{N₁ + N₉\}; GN₁₉ → N\{N₁ + N₉₉\}
    ...

...GN₉₉ → N\{N₉ + N₉₉\};
presents NP o. The problem, then, is to ‘transmit’
pears. Hence, V c is not known at the moment when
an unbounded number of embedded verbs before V c ap-
have to be sub-classified according to the type
begin with NP o. This noun phrase must be an acc-
ceptable object for the last verb, say V c, in the
sentence containing it, the verb V b must be sub-
classified according to type of subject, and V a
for oNPs (in 4) to be an acceptable object of
the verb V a that precedes, and also an accept-
subject for the verb V b of the embedded
sentences which satisfy verbal selection rules is
of Nj; I subdivide the sets S 1 pro
(6b), S 2,..., in the same way:
... etc.

I subdivide the sets S 1 pro
This new way of ordering the rules is the
basis for the sub-classification of verbs V 30,
take the object S 1. A verb V 30 accepts
only the sub-sets S i whose subject N j is an ac-
ceptable object for that verb. This is a selec-
tion rule between verbs: the verb V 30 selects an
object having a verb of a certain type 5.

The generation of recursively embedded sen-
tences which satisfy verbal selection rules is
now obtained as follows. First, let us choose a
rule developing the matrix sentence, for example
(15) N j t V 30 S 1
Now the verbs in the sub-class V 30 have been sub-
classified in the lexicon according to the type
of acceptable subject, N j and also according to
the type of acceptable complement S 1. By choos-
ing in (15) a verb in the sub-class (N j, S 1), I
obtain an acceptable sentence.

The selection between verbs mentioned here
has already been suggested by Z. Harris in the
framework of a system of sentence generation
based on the concept of the verb as an operator
acting on its arguments (approximately, its sub-
ject and object). Selection between verbs was
also used by M. Gross in order to account for
constructions like Je cours manger un gâteau, ?? Je cours attenter Max; here, the first verb
(of movement) selects for the type of verb that
can follow it.

§3, Elimination of the complex symbols

The schema (12) generates only acceptable
sentences; each verb in the lexicon is classifi-
cated according to which of the sub-classes defined
by (12) it belongs to; hence no verb will ever
appear in a schema of type (12) unless it is ac-
ceptable there. Then, since the process defined by
(10) is such that only acceptable nouns can be
chosen for the noun phrases GN (= N) in
these schemata, each schema must in fact give
rise to an acceptable sentence.

The situation is quite different, however,
for the rules containing oNP, NP o or NP d. These
cannot be developed as written, for two
reasons: (1) only noun phrases of the type N
are available, so that verbal selection rules
can be satisfied; (2) the syntactic functions
expressed by the subscripts on these noun phra-
escs can be obtained only by a sub-classification
of the verbs appearing with them. Thus, in order
to order the rules is the
basis for the sub-classification of verbs V 30,
take the object S 1. A verb V 30 accepts
only the sub-sets S i whose subject N j is an ac-
ceptable object for that verb. This is a selec-
tion rule between verbs: the verb V 30 selects an
object having a verb of a certain type 5.

The generation of recursively embedded sen-
tences which satisfy verbal selection rules is
now obtained as follows. First, let us choose a
rule developing the matrix sentence, for example
(15) N j t V 30 S 1
Now the verbs in the sub-class V 30 have been sub-
classified in the lexicon according to the type
of acceptable subject, N j and also according to
the type of acceptable complement S 1. By choos-
ing in (15) a verb in the sub-class (N j, S 1), I
obtain an acceptable sentence.

The selection between verbs mentioned here
has already been suggested by Z. Harris in the
framework of a system of sentence generation
based on the concept of the verb as an operator
acting on its arguments (approximately, its sub-
ject and object). Selection between verbs was
also used by M. Gross in order to account for
constructions like Je cours manger un gâteau, ?? Je cours attenter Max; here, the first verb
(of movement) selects for the type of verb that
can follow it.

§1 Elimination of oNP s

I subdivide the sets S 1, S 2, S 3, ... (cf.4
and 5) into S i subsets, where 1 runs through
the 31 possible values of the subject N (which
replaces oNP s). These subsets then constitute a
classification of the schemata S 1, ... according
to the type of subject that is acceptable for
the verb of the schema:

§31 Elimination of oNP s

Now the verbs in the sub-class V 30 have been sub-
classified in the lexicon according to the type
of acceptable subject, N j and also according to
the type of acceptable complement S 1. By choos-
ing in (15) a verb in the sub-class (N j, S 1), I
obtain an acceptable sentence.

The selection between verbs mentioned here
has already been suggested by Z. Harris in the
framework of a system of sentence generation
based on the concept of the verb as an operator
acting on its arguments (approximately, its sub-
ject and object). Selection between verbs was
also used by M. Gross in order to account for
constructions like Je cours manger un gâteau, ?? Je cours attenter Max; here, the first verb
(of movement) selects for the type of verb that
can follow it.
Next, $S_1$ is developed, using the schema (13), by one of two types of rules:

1. $S_1 \rightarrow N_1 d e V_3 \ S_1^c$
2. $S_1 \rightarrow N_1 d e V_3 \ N_0$

If rule 1 is chosen, another sentence is embedded, and a verb $V_3$ in the sub-class $(N_1, S_1^c)$ is chosen from the lexicon. But if rule 2 is chosen sentence embedding terminates with that rule.

The same method can be used for generating acceptable relative clauses $NP^{E2}$ (in 6). As an example, I rewrite one of the $NP^{E2}$ in terms of the noun phrases $N$:

(17) $NP^{E2} \rightarrow N_1 q u e \ N_3 t V_30 \ S_{pro1}$

By choosing a verb $V_30$ in the subclass $(N_3, S_{pro1})$, i.e., one taking $N_3$ as its subject and as second verb (in $S_{pro1}$), one whose subject is $N_1$, I guarantee that the $N_1$ in $a$ is both an acceptable object of $V_30$ and an acceptable subject of the verb in $S_{pro1}$.

§32 Elimination of $NP_0$

The development sketched in §31 will not do for relative clauses like $NP^{E2}$ (in 7), which have the form $NP_0 q u e S_{pro, i}$. This can be schematized roughly as $NP_0 q u e V_1 \ldots V_c$, where $V_1, V_2, \ldots$ are embedded verbs of the type $V_30, V_31, \ldots$, and $V_c$ is the last verb of $S_{pro, i}$, the one for which $NP_0$ must be an acceptable object.

In order to transmit the selectional characteristics of $NP_0$ to the rule that develops $V_c$, and this within the framework of a CF grammar, I can proceed as follows. I subscript $S_{pro}$ in (7) by $k$, which is also the subscript on the noun phrase $N_k$ that replaces $NP_0$ (just as $S_1, S_2, \ldots$, were subscripts for the type of subject); then the schema $S_{pro1}$ for embedded sentences will have two subscripts: one for $k$, and a second one for the type of subject the verb takes. This yields the following kind of development:

(18) $NP^{E2} \rightarrow N_k q u e S_{pro, k}$

$S_{pro, k} \rightarrow N_1 t V_3 (\theta)_{30}$
$\ldots$

If rule 1 is chosen, sentence embedding terminates; then, choosing a $V_3$ in the sub-class taking an object of type $N_k$ (as indicated by $\phi_k$) guarantees that $N_k$ in (18) is an acceptable object for that $V_3$. If rule 2 is chosen, sentence embedding continues; a verb $V_{32}$ is chosen, in the sub-class $(N_1, S_{pro, 31})$, until a rule of type b is chosen.

The reader will notice two features of this method of using the selection rules to generate relative clauses:

1. The subdivision of $S_1$ into a set of $S_{pro}$ schemata does not increase the number of rules in $S$. The same number of rules would be obtained by inserting the noun phrases $N_k$ into $S$ (or $S_1$), and this must be done in any case in order to express the verbal selection rules (in whatever fashion). In the decompositions of $S_1, \ldots$, used above, the point was only to present the original schemata so as to make the subject or object of the verb in the schema stand out, for further reference.
2. The two kinds of selection made explicit in these schemata, the one between verbs, and the other (better known) between verb and object (or subject), appear only once in the grammar. Both types of selection are used in each step of sentence embedding, but in no case does this entail rewriting the two kinds of selection in the grammar each time a deeper level of embedding is attained.

§4. Conjunction; respectively

It has been shown by Chomsky that conjunctions can be described in a CF grammar only by using an infinite number of rules, represented by rule schemata; if one restricts oneself to strict CF grammar, one introduces an excessive structuring of the conjoined forms. An approximate solution can nevertheless be given to this problem, in the framework of a finite CF grammar, in the following way. I construct a sequence of conjoined noun phrases:

(21) a. $GN_1 + N^c$; b. $GN_2 + N^c$ at $N^c$; c. $GN_3 + N^c$ at $N^c$ at $N^c$; d. $GN_4 + N^c$ at $N^c$ ... at $N^c$ (i times)
Although these strings cannot be generated by a CF grammar, the CF grammar containing the rules $G_{cf}^1$, $G_{cf}^2$, ..., $G_{cf}^n$, I can set up the series of grammars $G_{cf}^1$, $G_{cf}^2$, ..., $G_{cf}^n$, each representing a better approximation to the infinite grammar $G_{cf}^\infty$ which contains a noun phrase of unbounded CF length.

For any practical purpose, such as generation (or analysis) of sentences, it is clear that one of the $G_{cf}^n$ will be large enough to yield the desired precision. However, another approximation is available which is less costly, from the viewpoint of the number of rules required, and which yields the same result for $G_{cf}^\infty$. This is the rule schema proposed by Chomsky & Schützenberger for handling conjunction in a CF grammar. For the case of noun phrase conjunction, this schema is as follows:

\[
(22)\ a \ G_N \to N^* \quad b \ G_N \to N^* (\text{et} \ N^*)
\]

The star indicates that the group (et $N^*$) can be iterated as many times as is necessary. This schema is therefore an abbreviation for an infinite number of rules.

With such a rule schema in it, my grammar is no longer strictly CF; however, it is clearly faithful to the spirit of the approximation for $G_{cf}^\infty$ outlined above, since the language described by my grammar is the same as that reached asymptotically by the series of grammars $G_{cf}^1$, $G_{cf}^2$, ..., $G_{cf}^\infty$ obtained with (21). The rule schema (22) can be compared to an algorithm for generating any one of the grammars $G_{cf}^n$ by choosing the number of iterations.

There exists a set of structures in natural language which cannot be described by the methods developed until now, namely those containing either respectful or, the distributives qui or selon que:

\[
(23)\ a \ Les \ vues \ des \ groupes \ A \ et \ B \ réussissent \ et \ échouent \ dans \ les \ labyrinthes \ L_A \ et \ L_B \ respectivement.
\]

\[
b \ Les \ reportages \ ont \ parti \ qui \ aux \ minières, \ qui \ aux \ diététistes, \ qui \ aux \ députés.
\]

\[
c \ Selon \ que \ tu \ es \ pauvre, \ bourgeois \ ou \ aristocrate, \ tu \ seras \ ouvrier, \ commerçant \ ou \ patron.
\]

Although these strings cannot be generated by a CF grammar, a procedure is nevertheless available for including this type of sentence in the CF approximation under discussion here.

I add Kleene rules to the grammar, and a condition on these rules, as follows:

\[
(24)\ a \ N_s (\text{et} \ N) V N_o (\text{et} \ N)\]

\[
b \ N_s (\text{et} \ N) V (\text{et} \ V) N_o (\text{et} \ N)\]

These rules contain all common conjunctions of subject, verbs and direct object. Moreover, they cover the sequences of classes observed in sentences containing respectivement. They don't have the structure one would like to associate with such sentences. In order to describe the respectivement sentences, I add the following condition to the starred parentheses: the number of iterations of each occurrence of the star is the same; and a structure, or rule of interpretation, is imposed on the starred groups, as follows:

\[
(25) \ N_s (\text{et} \ N) V (\text{et} \ V) N_o (\text{et} \ N)\]

This grouping pairs the $N_s$ and the $N_o$ that are to be associated with each other via respectivement; (25) is equivalent to:

\[
(26) \ N^1 \text{et} \ N^2 \text{ et} \ V^1 \text{et} \ V^2 \text{ et} \ N^3 \text{et} \ N^4
\]

Thus, I am interpreting (25) as a sentence conjunction: $N^1 \text{et} N^2 \text{ et} V^1 \text{et} V^2 \text{ et} N^3 \text{et} N^4$, as required by the adverb respectivement.

§5. Conclusions

The methods I have sketched here can certainly be applied to other natural languages and will account in a natural way for the general phenomena of verbal selection rules in embedded sentences. One may wonder why this work has not been carried out before.

Historically, attacks against the adequacy of CF grammar for describing natural language arose at a moment when it was necessary to explore the nature of the transformational grammar just proposed. This new style of grammar seemed so much better adapted than CF phrase structure grammars to explaining sentence relations that any more effort towards developing a detailed CF grammar seemed fruitless. To discourage such efforts, Chomsky (chap. 5) declared that "any grammar that can be constructed in terms of this theory [CF phrase structure grammar] will be extremely complex, ad hoc and 'unrevealing'". These remarks were reaffirmed (Chomsky) and bolstered by an argumentation based on the inherent inadequacy of CF grammar for describing verbal selection rules.

A second criticism arose from the analysis of constructions, like respectivement, whose description could not be obtained within the strict framework of a CF grammar. We have seen above that such a statement is at best unclear. It may be correct that a mathematically rigorous description of this construction is not possible in a...
form of sentences containing respectively, is now used in recent work in generative semantics. The statement of the transformation can between sentences, but it does not contain the operations or the metalinguistic assertions that make the transformation explicit. By a small extension of the CF framework I can also obtain the equivalent of a transformation, as follows. As an example, I consider the passive transformation.

The passive transformation consists in matching an active phrase with its passive counterpart. The statement of the transformation can stop there, as does Harris's', or one can add the specification of the computer operations needed to create the active and passive trees, as in generative grammar. In the CF grammar presented here, I have two independent rules, one for the active form, and another for the passive of the first:

(27) a $S_{act} \rightarrow NPs \ t \ V2 \ NPo$

b $S_{pas} \rightarrow NPo \ t \ atäre \ V2ϕ \ (par \ NPs)$

Each of these rules has an independent set of selection rules that are expressed in the choice of the $S$ for the NP. Adding these selection rules, (27) becomes:

(28) a $S_{act} \rightarrow NPs \ t \ V2 \ NPo; NPs \rightarrow N_i; NPo \rightarrow N_j$

b $S_{pas} \rightarrow NPo \ t \ atäre \ V2ϕ \ (par \ NPs); NPo \rightarrow N_i; NPs \rightarrow N_j$

This is of course a wasteful repetition of identical selection rules. It was just to avoid this kind of useless duplication that justified the introduction of transformations. Suppose now that I factorize the selection rules from a set of forms that constitute an equivalence class, for example, from the 'active' and the 'passive' forms; I place a separator $ϕ$ between the forms of the equivalence class:

(29) $S \rightarrow NPs \ t \ V2 \ NPo/ϕ/ NPo \ t \ atäre \ V2ϕ \ (par \ NPs)/ϕ/ NPo \rightarrow N_i; NPo \rightarrow N_j$

In this formulation, the selection rules are no longer duplicated; moreover, we can interpret the separator $ϕ$ between the members of the equivalence class as indicating a relation between the sentence schemas so separated. The factorization of the selection rules, together with the introduction of the separator can be read as the definition of a transformational rule between the sentence schemata.

Of course, rule (29) is no longer CF, but it represents a rather natural extension of the CF framework which makes the latter much more similar to a transformational grammar than one might have thought possible up till now. However, the reader will note that the concept of a transformation is indispensable as a tool for the construction of this CF grammar, and then for its extension towards a transformational grammar by means of the factorization of selection rules. Furthermore, this CF grammar does not generate the sentences of the language 'weakly', in the meaning given this word by Chomsky; in fact, it provides them an 'adequate' grammatical structure as well as a linguistically justifiable relationship to other sentences of the language.

Finally, let us note that although the entire set of rules of the CF grammar proposed here is large (of the order of $10^9$ rules), it is nonetheless finite. Furthermore, the size of the grammar is of no theoretical consequence, since it could be stored in digitalized memory (e.g., a pile of discs), but in a dynamic form (that is, in the form of schemata) where each rule is generated at the moment when the program of syntactic analysis (or generation) requires it. In this way, the set of rules would be reduced to a series of sub-programs that can generate either one rule, or a sub-set of rules, or all the rules. During analysis or generation, a call for rules would activate their synthesis by the appropriate sub-program.

Such a program of analysis by synthesis reduces the number of rules to a smaller number of sub-programs, but a string grammar reduces them still more, down to a set of about 150 strings (the rewrite rules) together with about 200 restrictions (the CS portions attached to the CF rules).

The size of the CF grammar required to describe selection rules adequately also explains why all attempts at automatic syntactic analysis by means of strictly CF grammars undertaken until now have failed. The authors of these CF grammars limited their effort to including some rudimentary linguistic facts; the average size of this sort of CF grammar was of the order of several thousand rules (cf. Kunogi20). Under these conditions, there was no question of providing only linguistically acceptable analyses. However, in the last few years, other CS variants of a CF grammar have been proposed, and partly worked out. In particular, the augmented transition network grammar of Bobrow & Fraser2, especially in the form given it by Woods26, has predicates associated with the transitions, predicates that are so many context-sensitive tests. This kind of
grammar is then quite similar to string grammar, i.e., to a CF grammar together with CS conditions on the rules. Unfortunately, none of the grammars based on the ideas of Bobrow and Woods has been worked out in sufficient detail to make a linguistic comparison with string grammar possible.

BIBLIOGRAPHY

1. Bar-Hillel & Shamir, E., 1960. Finite-State Languages, in Language & Information, New York, Addison-Wesley, (1964).

2. Bobrow, D. & Fraser, B., 1969. An augmented state transition network analysis procedure, Proc. of the International Joint Conference on Artificial Intelligence.

3. Boons, J.-P., Guillet, A. & Leclere, C., 1976. Classes de constructions transitives, Rapport de Recherche N° 6, L.A.D.L., Univ. de Paris 7, Place Jussieu, Paris

4. Chomsky, N., 1955. The logical structure of linguistic theory, New York, Plenum (1975).

5. —— 1957. Syntactic Structures, The Hague Mouton

6. —— 1963. Formal properties of grammars, in Handbook of Mathematical Psychology, Vol. 2, New York, John Wiley

7. —— 1965. Aspects of the theory of syntax, Boston, MIT Press

8. —— 1966. Topics in the theory of generative grammar, in Current Trends in Linguistics, Vol. 3, The Hague, Mouton

9. Chomsky, N. & Schützenberger, M., 1963. The algebraic theory of context-free languages, in Computer Programming and Formal Systems, Amsterdam, North-Holland

10. Gross, M., 1968. Grammaire transformationnelle du français: le verbe, Paris, Larousse

11. —— 1972. Mathematical Models in Linguistics, New Jersey, Prentice-Hall

12. —— 1975. Méthodes en Syntaxe, Paris, Hermann

13. Harman, G., 1963. Generative grammars without transformation rules: a defense of phrase structure, Language, Vol. 39, N° 4.

14. Harris, Z. 1952. Discourse analysis, Language, Vol. 28, N° 1

15. —— 1962. String analysis of sentence structure, The Hague, Mouton

16. —— 1964. The elementary transformations, in Harris, 1970, Papers in structural and transformational linguistics, Dordrecht, Reidel

17. —— 1968. Mathematical structures of language, New York, John Wiley

18. Joshi & Levy, 1977. Constraints on structural descriptions: local transformations, SIAM J. of Computing, Vol. 6, N° 2

19. Kuno, S., 1963. The multiple-path syntactic analyzer for English, Report N° NSF-9, Computation laboratory, Harvard, Boston.

20. —— 1965. The predictive analyzer and a path elimination technique, Comm. of the ACM, Vol. 8, p. 453

21. Peters, S. & Ritchie, R., 1969. Context-Sensitive immediate constituent analysis, Proc. of the ACM Symposium on 'Theory of Computing', New York, ACM

22. Postal, P., 1964. Limitations of phrase structure grammars, in The structure of language, ed. by Fodor & Katz, New Jersey, Prentice-Hall.

23. Sager, N., 1973. The string parser for scientific literature, in Natural Language Processing, ed. by R. Rustin, New York, Algorithmics Press.

24. Salkoff, M., 1973. Une grammaire en chaîne du français, Paris, Dunod

25. —— 1979. Analyse syntaxique du français: grammaire en chaîne, Amsterdam, J. Benjamins

26. Woods, W., 1970. Transition network grammars for natural language analysis, Comm. of the Assn. for Comp. Mach., Vol. 13, p. 591

*I should like to thank M. Gross for many helpful comments, and myself for an excellent typing job.*