METARULES AS META-NODE-ADMISSIBILITY CONDITIONS

Technical Note 304

September 1983

By: Susan U. Stucky, Consultant

Artificial Intelligence Center
Computer Science and Technology Division

SRI Project 3560

The research reported herein was supported by the
National Science Foundation under Grant IST-8103550.

333 Ravenswood Ave. • Menlo Park, CA 94025
(415) 326-6200 • TWX: 910-373-2046 • Telex: 334-486

METARULES AS
META-NODE-ADMISSIBILITY CONDITIONS

Susan U. Stucky

Abstract

Metarule phrase-structure grammars (MPS grammars) have been shown to be an extremely powerful formalism in need of constraints from both the computational and the linguistic points of view. One problem with the standard generative interpretation of metarules is the generation of infinite rule sets. Furthermore, even if grammars having this property are disallowed, the possibility of a combinatorial explosion of rules still remains. In the present paper we explore a view of metarules as meta-node-admissibility conditions (MNACs) which allows a non-generative interpretation of metarules. Under such an interpretation, an MPS grammar will not have either of the two problems mentioned above. We find that, under one suggested implementation, the above mentioned problem appears under another guise, so that additional constraints are needed to ensure an effective procedure for checking admissibility conditions in the computational setting. The important observation is that one can, by parsing with MNACs on the fly, recognize languages for which the generative interpretation is not available.

1. Introduction

Metagrammatical formalisms of various types have been receiving increasing attention from theoretical linguists.[*] One such formalism is the metarule, a device for deriving one set of syntactic rules from another. The intuitive appeal to computational and noncomputational linguists alike lies in the concise statement of linguistically significant generalizations across sets of rules and the consequent reduction of the number of rules the grammar writer needs to formulate. Metarules have been most fully explored within the framework of Generalized Phrase Structure Grammar (GPSG).[1] In that formalism, metarules are rule templates used to derive phrase structure rules from other phrase structure rules. One could, for example, write a metarule for relating passive-verb-phrase rules to active ones. The form such a metarule might take is shown in (1), which states that if there is a rule that expands a nonpassive VP as a V followed by a NP (and any other category), there will also be a rule that expands a passive VP as a V followed by everything as before except for the NP (which does not appear).

$$(1)\quad \underset{[\text{-pass}]}{\mathrm{VP}} \rightarrow \mathrm{V\ \ NP\ \ X} \quad \Rightarrow \quad \underset{[\text{+pass}]}{\mathrm{VP}} \rightarrow \mathrm{V\ \ X}$$

Although metarules, as originally conceived of in GPSG, were intended as a non-pernicious addition to context-free phrase structure grammars, Uszkoreit and Peters (forthcoming) have shown for a particular definition of metarule-phrase-structure (MPS) grammars that, with no further constraints, such grammars can encode any recursively enumerable language. This is clearly an undesirable result from the computational point of view. This consequence is likewise disadvantageous for linguistic theory–for example, under some assumptions, learnability is

[*] This research was supported by the National Science Foundation grant No. IST-8103550. The views and conclusions expressed in this document are those of the author and should not be interpreted as representative of the views of the National Science Foundation or the United States government. The author acknowledges the many helpful comments and discussion on this topic from Chris Culy, Stanley Peters, Jane Robinson, and Stan Rosenschein. Particular thanks go to Hans Uszkoreit, who, in the early writing stages, asked some crucial questions (and helped in the formatting), to Stuart Shieber, who clarified a number of the computational ramifications, and to Fernando Pereira, who provided a definition crucial to the value of the ideas presented in this paper.

[1] Cf. Gazdar (1982), Gawron et al., (1982), and Thompson (1982).

precluded (cf. Gold 1967). There are, however, two related but separate problems: (1) the potential combinatorial explosion of rules in the grammar; (2) the generation of infinite sets of rules. These problems arise independently of the excessive power of the formalism; they would exist even if the power of the formalism were restricted to the generation of context-free languages.

The presence of infinite rule sets can also be a complicating factor, both for computational implementation and for linguistic theory. Computers cannot, in finite time, generate an infinite set of rules and people do not, presumably, learn an infinite set of rules. But, because the consequences of a grammar's encoding infinite sets of rules are less clear in linguistic theory than in the computational environment, we concentrate on the computational aspects of the problems outlined above. We anticipate, however, that the comments made here will be relevant to those who develop a theory of mental representation incorporating some version of MPS grammars.

To address the two problems just mentioned, we introduce a reinterpretation of metarules as meta-node-admissibility conditions (MNAC). Since no rules are derived under this reinterpretation, there can be neither a combinatorial explosion of rules nor a derivation of sets of infinite rules by the grammar itself. Nevertheless, for these results to be beneficial in the computational environment, it will be necessary to parse with MNACs directly "on the fly". While we conclude that this move allows us to use the finiteness of the input to determine when derivations should cease in one set of cases, we also observe that certain problems are inherited from the generative approach necessitating further constraints.

2. Problems

In this section we consider three separate but related problems that arise in the computational implementation of metarules. Since many of the devices that figure in GPSG (e.g. feature co-occurrence restrictions and ID/LP notation) are irrelevant to the present discussion, we rely on the less complicated definition of MPS grammars defined by Uszkoreit and Peters (forthcoming). A natural way to implement such a grammar is to expand out all the rules encoded by metarule; therefore, our comments in this section pertain to MPS grammars in accordance with this method of implementation.

Three problems arise when metarules are implemented in this way. The first is a potential combinatorial explosion of rules. A metarule which affects n symbols per rule adds (at least) n times the number of rules in the grammar. Note that this potential is present even if infinite rule sets are disallowed.

A more serious problem than a combinatorial explosion of rules is that it is possible to write metarules that generate infinite rule sets—clearly an undesirable result. Consider a rule of the form in (2) below which enumerates, by repeated application of the metarule, an infinite number of verb phrase rules, each containing one more prepositional phrase than its predecessor.

(2) VP → V X ⇒ VP → V X PP

This rule expresses the fact that (in English) an indefinite number of PPs may occur in a given sentence. Rules of this sort need not engender greater than context-free power. For instance, the grammar in (3) generates only the regular language ab* even though it enumerates an infinite number of rules.

(3) S → A
A → a
B → b

S → A X ⇒ S → A X B

Nevertheless, the rules defined by a grammar like that in (3) cannot be precompiled since the metarule generates an infinite rule set.

The third and most serious problem of all is that, as noted in the introduction, MPS grammars are capable of generating all recursively enumerable languages. Some constraint is called for so that the grammars incorporating metarules will be computationally feasible.

3. A redefinition: metarules as meta-node-admissibility conditions

We have seen that metarules can be extremely powerful, increasing the number of rules in the grammar and generating infinite rule sets and/or non-recursive languages. One way of exploring possible constraints is to change the interpretation of metarules in order to see if some constraints are more perspicuous or more easily implemented under this implementation. Though other reinterpretations are possible, the one explored in the present paper considers metarules as complex node-admissibility conditions, called meta-node-admissibility conditions.[2] Viewed intuitively, these MNACs, situated alongside the spelled-out phrase structure rules, function as part of the grammar.

The reader is referred to the Appendix for a restatement of Uszkoreit and Peter's MPS grammars (forthcoming) in terms of node admissibility. The definition of admissible pair (Definition 3) is a recursive definition stating the following: (1) a node in a tree is admitted if it matches some base rule; (2) a node is admitted if that node matches the output of some MNAC whose input is itself admissible.

3.1 An example

As an example of how this definition works, consider the grammar in (4) below. Under the standard metarule interpretation, the MNAC in (4b) would have generated an infinite set of phrase structure rules. Under the MNAC interpretation however, no infinite set of rules is generated and, significantly, a finite input will be analyzed by a finite number of steps when the definition is applied.

[2] The reader is referred to Shieber et al. (1983) for a catalog of possible formal constraints on metarules and the computational and linguistic ramifications of each such constraint.

(4) a. Base rules

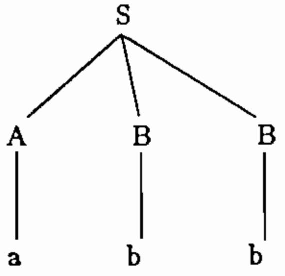
$S \rightarrow A$
$A \rightarrow a$
$B \rightarrow b$

b. MNAC

$S \rightarrow A \;\; X \quad \Rightarrow \quad S \rightarrow A\,X\,B$

We ask if, according to the definitions of node admissibility and the grammar in (4), the tree in (5) is admitted.

(5)



The nodes A and B are admitted by the relevant base rules directly (i.e., by Definition 3 Clause 1). We then ask if the S node is admitted. It is not admitted by a hase rule directly, but rather by the MNAC together with the base rules. The local set of nodes S → ABB[3] is admitted by Clause 2 of Definition 3, because it matches the output of the single MNAC, S → AX, under an input to the MNAC S ⇒ AB. This is admitted by virtue of the single MNAC once again. We arrive at the pair, S → A, analyzed as an input (S → AX, where X ranges over the null string), which matches the first base rule and is thus admitted by Clause 1 of Definition 3. Hence the tree in (5) is admitted, since all local sets of nodes are admitted.

3.2 Consequences of the definition

The solution to two of the problems listed in Section 2 is straightforward. Because

[3] In this loose notation, we use S → ABB to signify what is more properly notated as the pair <S,<A,B,B,>>

no rules are generated by metarule under this reinterpretation, no combinatorial explosion can result and no infinite rule sets can be derived. The reinterpretation leaves the weak generative capacity unchanged, so that any problems stemming from the power of the formalism are inherited.

As mentioned above, the most natural implementation of the MNAC definition is one that utilizes MNACs on the fly.[4] However, two conditions must be met to provide an effective procedure for recognition. First, because the definition is stated in terms of already existing tree structures, some mechanism independent of the grammar must be given to assign tree structures to input strings. This mechanism must ensure that the set of trees be finite and include (at least) those trees admitted by the base rules along with the MNACs. An algorithm of this sort, which generates tree structures for a class of trees without cyclic or empty non-terminals has been defined by Pereira (personal communication), and it appears that this mechanism can be further generalized to incorporate empty non-terminals. Second, we require a procedure that checks node admissibility and, crucially, that terminates in all cases. It is this second condition which is problematic.

To understand exactly why it is that the present definition would allow non-termination, consider the following three cases. The first case is a simple looping whereby one arrives at the same rule again and again. The addition of the MNAC in (6) to the grammar in (4) will have this effect.

(6) $S \rightarrow A\ B\ X \;\Rightarrow\; S \rightarrow A\ X$

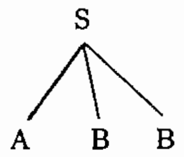
[4] Two discussions of parsing with metarules directly antedate the present discussion, Thompson (1982) and Kay (1982). Rosenschein (personal communication) has also explored the possibility. Both Thompson and Kay dismiss as impractical parsing with metarules on the fly. Thompson opts for precompilation of the rules generated by metarule and, in order to avoid generating infinite sets of rules, adopts finite closure of the rule set under the once-through hypothesis. Kay reinterprets metarules so that their application is instigated by a transducer situated between the base grammar rules and either the generator or the parser; his approach, however, in contrast to the one described here, requires ordering. These differences preclude any real comparison of their approaches with the present one.

There is one derivation in which the analysis of the second MNAC succeeds in deleting the symbol B, which has been added by the first MNAC. We arrive at the same rule again and again. This sort of looping (which is not restricted to rules that add and delete symbols but may occur with MNACs that reorder or rename symbols) can be detected easily. For each time that a derivation is conducted, we merely record all the rules arrived at and check whether such a rule has been derived before.

In addition to these looping derivations, there are two other types of recursion among metarules. One is constrained, the other not. The constrained type is that sort of recursion for which, given the input, it is possible to determine when to terminate the derivation. Consider, for instance, cases in which metarules only add symbols on the right-hand side, symbols which are not present on the left (and which are non-epsilon deriving) as in the grammar in (4)). The number of times the MNAC is "applied" is bounded by the length of the input. Therefore, given the input, there is an a priori bound on computation.

A similar, though not identical, case is illustrated by a derivation allowed by the grammar in (4a) with only the MNAC in (6), (which, remember, has symbols on the left-hand side not present on the right). Suppose one were analyzing the node in (7) below. If one continues to check the MNAC in (6), without making use of the number of symbols in the input, there will be no end to the derivation, since each rule arrived at is a new one (differing from the preceding one only in the addition of a single symbol). (Note that this version of the halting problem may arise even in a grammar that analyzes the regular language ab*. The problem is ultimately due to the properties of the formalism itself, even though one can write particular grammars that analyze only regular languages.) The pairs arrived at are different from any of the previous ones (differing from them in the addition of a single symbol). However, one may take the number of symbols in the input together with the MNAC in question to place an it a priori bound on computation.

(7)



Here, then, are two cases for which we can utilize the fact that the input we shall be analyzing (i.e. a string of words in a natural language) is available at the same time that the metarules are invoked. We cannot precompile all the rules that would be generated by such a metarule, but, by parsing with the same rule interpreted as an MNAC, we can analyze the finite strings that are admitted by it along with the base rules. Note that nothing prevents one from making use of the same information under the standard metarule interpretation, though doing so would, in effect, be applying the insights of the MNAC interpretation.

The ability to analyze such structures seems beneficial in the linguistic setting, since the structures admitted by this class of MNACs are among those attested in natural-language analysis. These structures are those in which an indefinite number of categories of the same type may occur as sisters (e.g., prepositional phrases and adverbs in many languages, noun phrase causees in flexible-word-order languages such as Japanese).[5]

A second linguistically valid result of allowing MNACs to apply more than once in the derivation of a rule is that rules which function to liberate symbols from some one rule into another rule typically have this property (cf. Pullum (1982)). These "liberation rules", which have the effect of admitting flat structures, are motivated for languages in which there

[5] This class of structures consists of those disallowed by the constraint on metarules that permits them to be applied only once in the derivation of any rule (Gazdar and Pullum 1982). In addition, it should be noted that, although the Kleene star notation allows for specifying as sisters an indefinite number of categories of the same type, the present analysis would produce different predictions. The symbols subsumed under the Kleene star would not be affected individually by other rules in the grammar—unless, of course, the Kleene star notation itself is regarded as a metagrammatical abbreviatory device that undergoes full expansion in the grammar prior to the application of certain other rules. But then we find ourselves back where we started from, i.e., at the expansion of some metagrammatical device into an infinite set of rules or symbols. There are other, more subtle differences. For instance, MNAC analyses—but not the Kleene star notation—would allow for the addition of two (or more) mutually dependent symbols in a particular rule. This could be useful in the analysis of causatives that often require matching of some specific symbol (e.g. a noun phrase) with another (e.g., a causative marker).

is evidence for continuous syntactic constituents present in some orders but not in others (e.g. Makua (Stucky (1983)).

Both of these linguistic consequences seem to play a role in an analysis of the Dutch cross-serial dependencies, a construction that has received considerable attention in the literature. (Recent papers include Bresnan et al. (1982) Culy (1983), Pullum and Gazdar (1982) and Thompson (1983).). A complete analysis of the data certainly is beyond the scope of this paper, but among the properties of interest with regard to the present issues are two-fold. First, the Dutch constructions permit an indefinite number of pairs of verbs plus their complements, and, second, the noun phrase complements appear in any order in a noun phrase sequence as do the verbs in the verb sequence. The syntactic structure to be assigned to these constructions is a matter of controversy, but some mechanism for ensuring the potentially indefinite number of complement-verb pairs is needed. An illustrative example taken from Bresnan et al. (1982:615) is the following.

(8) ...dat de leraar Jan Marie de kindereen leerde laten
that the teacher Jan Marie the children teach-past make-inf

lere zwemmen
teach-inf swim-inf

'...that the teacher taught Jan to make Marie teach the
children to swim'

In order to derive the linguistic benefits to be gotten by allowing rules that add symbols, we have to make sure that the grammar does not have the second sort of non-looping recursion—the type encountered in grammars with MNACs that allow the rules being derived to alternately grow unboundedly and shrink without our ever arriving at the same rule as before or without terminating. For instance, the grammar generating the classic triple-counting language $a^n b^n c^n$ in Uszkoreit and Peters (forthcoming) has that property.

Since determining whether a grammar has insidious recursion among the MNACs is itself undecidable, we must content ourselves with more restrictive properties. One way of constraining grammars is to disallow any shortening MNACs, i.e., any MNACs whose right-hand sides cover fewer symbols than do the left-hand sides. Because backchaining through the metarules requires adding symbols in the derivation for those MNACs that have shorter right-hand sides, it is this addition that presents the problem for the MNAC interpretation (just as it is the opposite case, the deletion of symbols that is problematic in the generation usage). If we adopt such a constraint, then some analysis of so-called deletion phenomena over and above the simple deletion of symbols in an MNAC would be necessary. Whether this last constraint is advisable on linguistic grounds is a question that warrants further investigation.[6] In any event, some such constraint is essential if an effective procedure for recognition is to be achieved.

4. Summary

Though the redefinition of metarules as MNACs does eliminate problems in grammar generation, i.e., by doing away with large and/or infinite rule sets, we have (not surprisingly) needed additional constraints on the form MNACs may take to enable a procedure for checking admissibility conditions in the computational setting and thereby gaining decidability. The important observation is that, by parsing with MNACs on the fly, we can make use of the input to allow recognition of languages for which the generative interpretation is not available. For instance, we can recognize languages with an indefinite number of symbols as sisters, when there is some linguistic dependency among those symbols. We have argued that this property is one that is exemplified by natural-language grammars and which, consequently, ought to be sanctioned by linguistically motivated formalisms.

[6] The effects of such a constraint on the weak generative capacity is likewise not investigated here.

References

Bresnan, J., R. M. Kaplan, S. Peters., A. Zaenen, 1982: " Cross-Serial Dependencies in Dutch" *Linguistic Inquiry*, volume 13, number 4, pp. 613-36.

Culy, C.D., 1983: "An Extension of Phrase Structure Rules and its Application to Natural Language." Masters thesis, Stanford University, Stanford, California (May).

Gawron, J. M., et al., 1982: "Processing English with a Generalized Phrase Structure Grammar," in *Proceedings of the 20th Annual Meeting of the Association for Computational Linguistics*, University of Toronto, Toronto, Canada (16-18 June).

Gazdar, G., 1982: "Phrase Structure Grammar," in P. Jacobson and G.K. Pullum, eds., *The Nature of Syntactic Representation* (Reidel, Dordrecht, Holland).

Gazdar G. and G.K. Pullum, 1982: "Generalized Phrase Structure Grammar: A Theoretical Synopsis," (Indiana University Linguistics Club, Bloomington, Indiana, August).

Gold, E., 1967: "Language Identification in the Limit", *Information and Control*, volume 16, pp. 447-74.

Kay, M., 1982: "When Meta-Rules are Not Meta-Rules," in M. Barlow, D. P. Flickinger, and I. A. Sag, eds., *Developments in Generalized Phrase Structure Grammar*, Stanford Working Papers in Grammatical Theory, volume 2 (Indiana University Linguistics Club, Bloomington, Indiana, November).

Pullum, G.K. and G. Gazdar, 1982: "Natural Languages and Context-free Languages", *Linguistics and Philosophy*, volume 4, number 4, pp. 471-504.

Shieber, S.M., S. U. Stucky, H. Uszkoreit, and J. J. Robinson, 1983: "Formal Constraints on Metarules," *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, Massachusetts Institute of Technology, Cambridge, Massachusetts (15-17 June), pp. 22-7.

Stucky, S. U. 1983: "Verb phrase constituency and linear order in Makua", in G. Gazdar, E. Klein, and G.K. Pullum eds., *Order, Concord, and Constituency*, Foris Publications, Dordrecht, Holland, pp. 75-94.

Thompson, H., 1982: "Handling Metarules in a Parser for GPSG," in *Proceedings of the 20th Annual Meeting of the Association for Computational Linguistics*, University of Toronto, Toronto, Canada (16-18 June).

Thompson, H., 1983 "Crossed Serial Dependencies: A low-power parseable extension to GPSG", in *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, Massachusetts Institute of Technology, Cambridge, Massachusetts (15-17 June), pp. 16-21. (June 1983) pp. 16-21.

Uszkoreit, H. and S. J. Peters, forthcoming: "On Some Formal Properties of Metarules".

APPENDIX

Definition of MPS Grammars under the MNAC Interpretation:[*]

Definition 1. MPS grammars:
An MPS grammar is a sextuple $\langle V_T, V_N, S, \rightarrow, V_V, \Rightarrow\rangle$ such that V_T and V_N are finite, disjoint sets of terminal and nonterminal symbols respectively, $S \in V_N$, V_V is a finite set of essential variables disjoint from $V_T \cup V_N$, $\rightarrow$ is a finite subset of $V_N \times (V_T \cup V_N)^*$ and $\Rightarrow$ is a finite subset of $(V_N \times (V_T \cup V_N \cup V_V)^*) \times (V_N \times (V_T \cup V_N \cup V_V)^*)$ such that if $\langle\langle A, \phi\rangle, \langle B, \psi\rangle\rangle \in \Rightarrow$ then each member of V_V occurs an equal number of times in ϕ and ψ and has at most one occurrence in each.

Let $G = \langle V_T, V_N, S, \rightarrow, V_V, \Rightarrow\rangle$ be an MPS grammar, bere and hereafter.

Definition 2. directly analyzes:
A pair $\langle A, \phi\rangle$ directly analyzes a pair $\langle B, \psi\rangle$ by a MNAC of G (in symbols $\langle A, \phi\rangle \Rightarrow \langle B, \psi\rangle$) if for some pair $\langle\langle A, \chi\rangle, \langle B, \omega\rangle\rangle \in \Rightarrow$ there are a positive integer n and strings $\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_{n-1}, \gamma_1, \ldots, \gamma_n, \delta_1, \ldots, \delta_{n-1} \in (V_T \cup V_N)^*$ and symbols $\varsigma_1, \ldots, \varsigma_{n-1}, \eta_1, \ldots, \eta_{n-1} \in V_V$ such that

(i) $\beta_i = \delta_j$ when $\varsigma_i = \eta_j$ $(1 \leq i, j < n)$

(ii) $\chi = \alpha_1\varsigma_1 \ldots \alpha_{n-1}\varsigma_{n-1}\alpha_n$

(iii) $\phi = \alpha_1\beta_1 \ldots \alpha_{n-1}\beta_{n-1}\alpha_n$

(iv) $\omega = \gamma_1\eta_1 \ldots \gamma_{n-1}\eta_{n-1}\gamma_n$

(v) $\psi = \gamma_1\delta_1 \ldots \gamma_{n-1}\delta_{n-1}\gamma_n$.

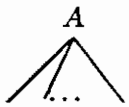
Definition 3. admission of a pair:
A pair $\langle A, \phi\rangle$ is admitted iff

(i) $\langle A, \phi\rangle \in \rightarrow$

(ii) there exist an MNAC M and a pair $\langle B, \psi\rangle$ such that $\langle B, \psi\rangle$ directly analyzes $\langle A, \phi\rangle$ by M and $\langle B, \psi\rangle$ is itself admitted.

[*] The definition given here is a restatement of Uszkoreit and Peters' (forthcoming) MPS grammar definition. Their statement defining an MPS grammar is taken directly and corresponds to my definition 1. My definition 2 is identical to their definition 1 with the exception that "directly yields" is replaced by "directly analyzes". The remaining three clauses are different. I thank Stuart Shieber for the five clause definition here, which replaces the more cumbersome six clause definition I started out with.

Definition 4. admission of a local set of nodes:
A local set of nodes



ϕ_1 ϕ_2 ... ϕ_n is admitted iff $\langle A, \langle \phi_1 \ldots \phi_n \rangle \rangle$ is admitted.

Definition 5. tree admissibility:
A tree is admissible iff for all local sets of nodes N, in a tree, each is admitted.