INDEXING AND REFERENTIAL DEPENDENCIES WITHIN BINDING THEORY: A COMPUTATIONAL FRAMEWORK

Fabio Pianesi
Istituto per la Ricerca Scientifica e Tecnologica
38050, Pante' di Povo - Trento - Italy
pianesi@irst.it

ABSTRACT

This work is concerned with the development of instruments for GB parsing. An alternative to the well known indexation system of (Chomsky, 1981) will be proposed and then used to formalize the view of Binding Theory in terms of the generation of constraints on the referential properties of the NPs of a sentence. Finally the problems of verification and satisfiability of BT will be addressed within the proposed framework.

1 Introduction

This work is concerned with the development of instruments for GB parsing (see Barton, (1984); Berwick (1987); Kolb & Tiersch, (1990)); in particular, our attention will be focused on the Binding Theory (henceforth, BT) a module of the theory of Government and Binding (henceforth, GB; see Chomsky (1981; 1986)). It has been pointed out (eg. in Kolb & Tiersch, (1990)) that the lack of a complete and coherent formalization of a linguistic theory like GB can be a major obstacle in addressing the issue of principle-based parsing; this is true of BT too, in particular with respect to the indexing system of Chomsky (1981), the shortcomings of which have often been pointed out in the literature. A formalism for the treatment of the referential relationships among the NPs of a sentence will be presented that is more expressive than indexation and more effective as a computational tool.

In Section 2 the indexing system and the role it plays within BT will be discussed; in Section 3, an alternative will be developed that overcomes some of the shortcomings of indexing. Such a system will, then, be used to formalize the view of BT as a device that generates (syntactic) constraints on reference. In Section 4, it will be shown how our proposal could be applied to some computational problems, i.e. the problems of verification and satisfiability within BT.

2 Preliminaries

Since Chomsky (1981), it has become commonplace to denote the interpretative relations among the NPs of a sentence by means of indices, i.e. integers attached to NPs in such a way that elements bearing the same index are taken to denote the same object(s), while different indices correspond to different denotations; most of the statements of BT have been laid down in terms of this system (Chomsky, 1981, 1986). In a number of works (see Chomsky (1981), Higginbotham (1983) and Lasnik & Uriagereka (1988)), however, it has been pointed out that the indexing device is not adequate to capture certain referential relations; this is true for the relation between pronouns and split antecedents, i.e. antecedents composed of two or more arguments bearing different thematic roles.[1] Furthermore, indices blur the distinction between coindexing under c-command and coindexing without c-command, thereby making it difficult to capture the dependence of an element, behaving like a variable, upon its antecedent (see Reinhart, (1983)).[2] The replacement of indices with index sets has been proposed as a way to address the first problem (see Higginbotham, (1983)): an ordinary index is substituted by a singleton; when there are pluralities, e.g. when an NP is coindexed with a split antecedent, it is annotated with the (set) union of the index sets of each component of the plurality; therefore, coindexing amounts to equating index sets. In this view, the ordinary conditions on disjoint reference (Principles B and C of BT) must be extended to avoid not only identical reference but, more generally, reference intersection. It has also been argued (Higginbotham, 1983) that indices should be abandoned and substituted by the non symmetric relation of linking; when the antecedent is split, a plurality of links should be used. This way, however, two different situations are collapsed together: the one in which an item is coindexed with a plurality of elements all of which share the same index, and the case of true split antecedents, where the elements composing the antecedent do not have the same index. Furthermore, the asymmetric behaviour of linking has no clear correlate at the structural level; it will be suggested below that c-command should continue to play a role here.

Computational works about BT have been mainly concerned with providing lists of possible or impossible antecedents for the NPs of a sentence (see Correa (1988); Ingria & Stallard (1989)); additional procedures select actual antecedents

---

[1] R-expressions can take split antecedents too, at least in certain cases (epithets); however, we will not explicitly address this point here. Anaphors, instead, can never take split antecedents.

[2] There is a full range of phenomena for which such a distinction seems crucial, eg. weak crossover and sloppy reading of pronouns (Reinhart, 1983); donkey sentences and the so called indirect binding (Haïk, 1984; Reinhart, 1987). However, only few of them will be addressed here.

among the potential ones. Berwick (1989) considers only R-expressions and a device (actually, a Turing machine) assigning the same index to multiple occurences of the same R-expression (names); furthermore, a set of disjoint indices is associated with each item. Finally, Fong (1990) performs a combinatorial analysis of the paradigm of free indexation, as proposed in (Chomsky, 1981); he shows that free indexation gives rise to an exponential number of alternatives and argues for the necessity of interleaving indexing and structure computation. In any case, indexing has been either explicitly or implicitly assumed, so that most of the computational approaches to BT suffer the same shortcomings pointed out above. In particular, given that both split antecedents and the distinction between binding and coreference cannot be easily accounted for, this results in an impoverished input being provided to the semantic (intepretative) routine.

In the following section a formal system will be discussed that tries to address such problems by explicitly distinguishing between binding and coreference; at the same time, BT will be seen as a theory that states very general constraints (constraint schemata), which are then (at least in part) instantiated according to the structural properties of the sentence at hand. These instantiated constraints are then used to test sets of positive specifications (indexations) which constitute the input to further semantic processing.[3]

3 The formal apparatus

For a given sentence w, let $N=\{n_1, n_2, ..., n_m\}$ be the set of its NPs; furthermore let us indicate with A, P and R the subset of N whose members are anaphors, pronouns and R-expressions, respectively. Sets A, P, R, constitute a partition of set N. Finally, Q denotes the set of quantified expressions and syntactic variables. Split antecedents will be considered as members of the power set of N, $\mathcal{P}(N)$; for the sake of uniformity, single NPs will be denoted by members of $\mathcal{P}(N)$ with cardinality equal to one, i.e. by singletons.

Definition 1 A relation $s \subseteq (\mathcal{P}(N) \times \mathcal{P}(N))$ is defined such that $(\phi\ \psi) \in s$ iff $\phi=\{m\}$, $\psi=\{n_1, ..., n_p\}$, $p>1$ and $m \in \psi$.

For any $\phi=\{n\}$, $n \in N$, sets $\mathcal{A}(n)$, $\mathcal{B}(n)$ and $\mathcal{C}(n)$ will denote the set of elements that c-command n and lie inside its binding domain whenever, respectively, $n \in A$, $n \in P$ or $n \in R$; finally, if n is a pronoun $\mathcal{D}(n)$ will denote the set of NPs c-commanding it and lying outside its binding domain.[4]

Definition 2 Given a sentence w, a relation $b \subseteq (\mathcal{P}(N) \times \mathcal{P}(N))$ is defined, such that $(\phi\ \psi) \in b$ iff one of the following conditions obtains:

(i) $\phi=\{n_i\}$, $n_i \in A$, $\psi=\{n_j\}$ and $n_j \in \mathcal{A}(n_i)$;

(ii) $\phi=\{n_i\}$, $n_i \in P$, $\psi=\{n_j\}$, and $n_j \in \mathcal{D}(n_i)$.

Definition 3 Given a sentence w, a relation $d \subseteq (\mathcal{P}(N) \times \mathcal{P}(N))$ is defined, such that $(\phi\ \psi) \in d$ iff $\phi=\{n_i\}$, $\psi=\{n_j\}$ and either $n_j \in \mathcal{B}(n_i)$ or $n_j \in \mathcal{C}(n_i)$, depending on whether $n_i \in P$ or $n_i \in R$.

In the following, $b_{(-)}$ and $s_{(-)}$, the inverse relations, will be used as well.

Definition 4 Given a sentence w and a phrase structure tree representation for it, τ_w, the set of *binding constraints* for τ_w is the set $\Re_w=\{(\phi\ r\ \psi) \mid \phi, \psi \in \mathcal{P}(N), r$ is a symbol, $r \in \{d, b, b_{(-)}\}\}$, such that $(\phi\ r\ \psi) \in \Re_w$ iff $(\phi\ \psi) \in r$, where r is the corresponding relation.[5]

Given sentence w and a phrase structure representation, a binding constraint set states disjoint reference constraints (essentially, the consequencies of Principle B and C of BT) and the range of the binding relation (see below) for each NP. The meaning of the formers is that whenever $(\alpha\ d\ \beta) \in \Re_w$, the intersection of the references of α and β is empty. Note that $\Re_w$ does not exhaust the range of possible constraints on reference; for instance, those preventing weak crossover violations or circular readings are not included in $\Re_w$ but will be discussed later on; furthermore, split antecedents are not mentioned in $\Re_w$

Let us, now, focus the attention on how to represent positive referential relationships. To this purpose, two fundamental relations on $\mathcal{P}(N)$, coreference and binding (more precisely, the bound variable reading, in the terminology of Reinhart (1983)) are introduced. The former is a transitive, symmetric and reflexive relation, therefore an equivalence relation; the latter is irreflexive, intransitive and non symmetric, it only obtains under c-command and denotes the dependence of an item upon another one for its interpretation.[6] An

[3]Disjoint reference constraints arising from Principles B and C of BT are not carried over to semantic routines but are resolved at an earlier stage. Furthermore, it is assumed that, whatever processing the semantic routines perform, their default behaviour consists of assigning non-sharing semantic import to different NPs, unless otherwise stated in the input constraint set.

[4]The relevant notion of *c-command*, is the following: node α c-commands node β in the tree τ iff α does not dominates β and every node γ dominating α also dominates β. In a sense, $\mathcal{A}(n_i)$, $\mathcal{B}(n_i)$ and $\mathcal{C}(n_i)$ are partial encodings of, respectively, Principles A, B and C of BT; see Giorgi, Pianesi, Satta (1990) for algorithms that compute these sets.

[5]Here, it is assumed that τ_w has been built according to all the modules of the theory, a part BT.

[6]Both binding and coreference are formal relations in that

item can be bound by, at most, one other element; on the contrary, an NP can corefer more than once and even with itself. Split antecedents cannot be bound and, finally, it is not possible for an item, α, to be bound and, at the same time, to corefer; on the other hand, α can be a binder and, at the same time, corefer. The binding relation will be denoted by the symbol *l*.

The differences between binding and coreference are at both the structural and the interpretative level. Binding can only obtain under c-command while this is not a prerequisite for coreference; at the interpretative level, the reference of the binder can be accessed to form the reference of the bindee. Instead, coreference corresponds to a sort of extensional identity and simply amounts to equating independent references; of course, items that do not refer (e.g., quantified expressions and anaphors) cannot corefer.[7] Bound items behave similarly, i.e. even a pronoun, when bound, loses the capability of autonomously referring and, therefore, of coreferring. Transitivity has not been assumed for binding, in order to avoid reducing the interpretation of a sequence of elements $a_1,..., a_n$, such that each a_i is bound by a_{i+1}, upon that of the last element; consider the following sentence:

(1) John and Mary told each other PRO to leave.

and the two readings:

(2) (i) John told Mary that Mary should leave and Mary told John that John should leave.

(ii)* John told Mary that John should leave and Mary told John that Mary should leave.

Because of obligatory control, *PRO* is bound by the reciprocal, which, in its turn, is bound by the matrix's subject. If binding were transitive, we should conclude that the interpretation of *PRO* is entirely dependent upon that of *John and Mary* (in this being on a par with the reciprocal) and the relevant reading would be (2.ii). However, (1) has only the first of the two readings in (2) and this requires that *PRO* inherits reciprocality from *each other*; therefore, the correct dependencies are between *PRO* and *each other* and between the latter and the matrix subject.[8] Note that a sentence like

(3) John and Mary told each other that they should leave.

admits both readings, given that the subject of the dependent clause can be bound either by the reciprocal or by the matrix subject. In this work, then, binding has a functional nature which may well reflect properties of semantic processing; even in this case, however, the point is that syntax only addresses an abstract property, i.e. functionality.

Since coreference is an equivalence, the representation could be simplified by considering a minimal relation corresponding to coreference. The connected parts of the graph of the coreference relation are complete subgraphs; for each of them, A=(V, E), choose an arbitrary vertex, α, and consider the graph $A_{min}=(V, \{(\beta\ \alpha) \mid \beta\neq\alpha, (\beta\ \alpha)\in E\})$. By iterating the procedure and then taking the union of the results, a (directed) graph is obtained that represents the minimal relation corresponding to coreference.[9] We will denote such a minimal relation with the symbol *c* and call it 'coreference' *tout court*. The inverses of both *l* and *c*, $l_{(-)}$ and $c_{(-)}$ will be used as well.

At this point, the notion of indexation set can be defined.

Definition 5 A *indexation set* for a sentence *w* is the set $\Im_w=\{(\phi\ r\ \psi) \mid \phi, \psi\in \mathcal{P}(N)$, *r* is a symbol and $r\in \{c, c_{(-)}, l, l_{(-)}, s, s_{(-)}\}\ \}$ such that $(\phi\ r\ \psi)\in\Im_w$ iff $(\phi\ \psi)\in r$, where *r* is now interpreted as the corresponding relation.

Note that split antecedents (relation *s*) are seen as part of the indexation set of the sentence since they do not have any independent status within syntax; furthermore, this move permits us to only consider a limited number of them every time, instead of the exponential number of possible split antecedents arising by free combinatorics.

3.1 Compatibility of an indexation set with BT

An indexation set is composed of positive specifications that interpretative procedures process in order to assign actual references. Before this could happen, however, it must be verified that each of such specifications does not contradict the sentence particular constraints of $\Re_w$ and general BT restrictions. In other words, a way is needed to enforce the overall compatibility of $\Im_w$ w.r.t. BT.

A *path* in $\Im_w$ is a sequence of elements $p=(\phi_0\ r_1\ \phi_1)\ (\phi_1\ r_2\ \phi_2)...\ (\phi_{m-1}\ r_m\ \phi_m)$, $m\geq 1$; if $\phi_0=\phi_m$

they are largely determined by structural properties. No pragmatic import is assumed for coreference, as is done by Reinhart (1983).

[7]See Haïk (1984) for a discussion about the distinction between referring and non referring NPs.

[8]Here, it is assumed that a VP containing a reciprocal, e.g. *told each other*, is true of each element *a* such that *a* is in the interpretation of *each other* and *told(a, b)* is true, where *b* is also in the interpretation of *each other* and $a\neq b$; see Higginbotham (1981, 1985).

[9]No information is lost in the passage from coreference to its minimal counterpart; the original graph can, in fact, be easily recovered by reintroducing transitivity, symmetricity and reflexivity. Of course, the choice of α does not affect the result.

then p is a *circular path*. Furthermore, the string $w_p=r_1\ r_2\ \ldots\ r_m$ is called the *path string* associated with p. Path strings may be used to define the following regular languages that will be useful to state many of the conditions about index sets in a compact form: $L_1=l^*(c+c_{(-)}+cc_{(-)}+c_{(-)}c+l+l_{(-)})l_{(-)}^*$, $L_2=\{s\}+\{s\}L_1+L_1\{s\}+L_1\{s\}L_1$, $L_3=\{s_{(-)}\}+\{s_{(-)}\}L_1+L_1\{s_{(-)}\}+L_1\{s_{(-)}\}L_1$. Let us briefly discuss their meaning. The paths from an element, ϕ, to another one, ψ, with strings in L_1 encode all the possible ways in which ϕ and ψ can be related by a combination of binding and coreference relations (in such a way, of course, that their definitory properties are respected). In this respect, L_1 replaces the traditional notion of coindexation (although we will continue to use this (improper) term to denote the existence of a path with string in L_1). Therefore, given a sentence like the following one (where subscripts are only used to single out constituents):

(4) $\quad$ His$_1$ mother told John$_2$ that he$_3$ should leave

a possible indexation set may contain the following elements: $(3\ l\ 2)$, $(2\ c\ 1)$ and the string lc for the path from 3 to 1, may be taken to substitute the old notion of coindexation. Consider, now, the notion of *referential contribution*; the basic case is given by the configuration $(\phi\ s\ \psi)\in\Im_w$ (i.e., an element contributing to a split antecedent); by extension, language L_2 encodes all the cases in which an element contributes to the reference of another one. For instance, a possible indexation set for the following sentence

(5) $\quad$ John$_1$ told Mary$_2$ that they$_3$ should leave

is $\{(1\ s\ 4), (2\ s\ 4), (3\ l\ 4)\}$; in this case, 1 and 2 are both contributing to the reference of 4 (the split antecedent) and of 3. On the other hand, language L_3 encodes all the cases in which an element ϕ receives a referential contribution from ψ. Finally, consider *overlapping reference* between two items; the basic instance is given by two split antecedents some component of which are either shared or coindexed; the general configuration gives rise to paths with strings in the language L_3L_2, the concatenation of L_3 with L_2.[10] An example is the following sentence:

(6) $\quad$ John$_1$ told Mary$_2$ that they$_3$ should avoid telling Henry$_4$ that they$_5$ had been discovered

with the following indexation set: $\{(1\ s\ 6), (2\ s\ 6), (1\ s\ 7), (4\ s\ 7), (3\ l\ 6), (5\ l\ 7)\}$. In this case, two split antecedents (6 and 7) are introduced that share the component 1; therefore, overlapping reference obtains between 6 and 7 and between 3 and 5.

The BT version considered here consistes of Principles A, B and C, as given by Chomsky (1986), weak crossover (see Reinhart (1983)) and some restrictions on circular readings. Now we can state the following:

Theorem 1 - Conditions for the compatibility of an index set with BT Given a sentence w, a tree representation τ_w and the binding constraint set, $\Re_w$, an index set, $\Im_w$, complies with BT iff the following statements hold:

(i) for any pair $\phi=\{n_i\}$, $\psi=\{n_{j1}, \ldots, n_{jp}\}$, $1\leq p$, if $(\phi\ l\ \psi)\in\Im_w$ then $(\phi\ b\ \psi_k)\in\Re_w$, $1\leq k\leq p$, where $\psi_k=\{n_{jk}\}$; i.e. binding relations cannot be arbitrarily introduced in $\Im_w$, but must be derived from the relation b in $\Re_w$.

(ii) for any $\phi=\{n\}$ there is no circular path in $\Im_w$, from ϕ, with string in l^+; i.e. there are no circular dependencies;

(iii) for any $\phi=\{n\}$, no circular path in $\Im_w$ gives rise to strings in L_2; i.e. an element is never coindexed with another one and, at the same time, contributes to its reference;

(iv) for any $\phi=\{n\}$, if $n\in A$ then there is a ψ such that $(\phi\ l\ \psi)\in\Im_w$ and $|\psi|=1$; i.e. each anaphor is bound in $\Im_w$ and never takes a split antecedent;

(v) for any $\phi=\{n\}$, if $n\in Q$ then there is no element ψ such that $(\phi\ c\ \psi)\in\Im_w$ or $(\psi\ c\ \phi)\in\Im_w$; i.e. quantifiers and syntactic variables cannot corefer; therefore, they can only function as binders;[11]

(vi) if $(\phi\ d\ \psi)\in\Re_w$ then there are no paths, in $\Im_w$, from ϕ to ψ with strings either in L_1 or in L_2 or in L_3L_2; i.e. if two elements are in a principle B or principle C configuration then: they cannot be coindexed; no one of them can contribute to the reference of the other and, finally, their references do not overlap.

This theorem states the conditions for the compatibility of an indexation set for a sentence w with BT. Note, that certain constraints, expecially those in (vi), make crucial use of the set $\Re_w$; other constraints, instead, directly apply to $\Im_w$, e.g. that

[10] In the linguistic literature, the term *overlapping reference* is used for all cases in which the reference of two items is not disjoint; of course, this implies that at least one of them denotes a plurality. However, the present use of this term, and that of referential contribution as well, is restricted to split antecedents, seen as the means, available to syntax, to compose pluralities and does not generalize to all the possible different varieties of plurals, such as those considered in (Lasnik (1976) and Higginbotham (1983)).

[11] (v) together with (i) enforces the ban against weak crossover in that (v) checks that no quantifier corefers and (i) only admits binding under c-command.

preventing weak crossover.

4 Applications

Two applications of the formalism introduced above are now considered. The discussion will by no means be exhaustive, the purpose being just to show the potentiality of the present proposal.

4.1 Verification

We define the verification problem for BT as follows: let τ_w be a phrase structure tree representation for a sentence w and let $\mathfrak{I}_w$ be an indexation set for w. We want to know if $\mathfrak{I}_w$ is compatible with the constraints imposed by BT on τ_w. In essence, this is the same problem as that discussed in the last section. A polynomial time algorithmic method that solves it will be briefly discussed. The problem at hand can be reduced to the following one: let R be a set of symbols and $G_R=(V, E)$ be a graph whose edges are triples $(\phi_1\ r\ \phi_2)$ where $r \in R$; given a regular language $L_R \subseteq R^*$, is there any path p in G_R with string in L_R? An algorithm can be given, based on a dynamic programming method presented in Aho et al. (1974), which takes as input one such graph G_R, a finite state non deterministic automaton for L_R and computes a $|V| \times |V|$ boolean matrix such that its i,j-th entry is set to 1 just in case there is a path, from node n_i to node n_j, with string in L_R.

The verification problem for BT can, then, be solved by the following algorithmic schema: first, compute relations d and b; then check condition (i) of Theorem 1 for every element in $\mathfrak{I}_w$. If the test is successful, build the directed labelled graph $G_V=(V, E)$ where $V=\{v \mid v \in \mathcal{P}(N)$ and either $(v\ r\ \phi) \in \mathfrak{I}_w$ or $(\phi\ r\ v) \in \mathfrak{I}_w$, for some r in $\{c, l, l_{(-)}, s, s_{(-)}\}$ $\}$ and $E=\mathfrak{I}_w$. Now, check conditions (ii) through (v) of Theorem 1, by means of successive runs of the algorithm previously sketched.

4.2 Satisfiability

Satisfiability for BT can be stated as follows: given a sentence w and a phrase structure tree representation for it, τ_w, does there exist at least one indexation set which is BT compatible? Observe that, addressing the problem of BT satisfiability can prove useful in actual parsing systems, since it provides a means to weed out ungrammatical analysis of the input string.

According to the version of BT considered here, only anaphors need to be considered; in fact, from the point of view of the syntactic theory, it is always possible to assign every R-expression and every pronoun an independent reference so that no interactions arise. In other words, a sentence like *She loves her* is perfectly grammatical, provided that the two pronouns are neither in the binding nor in the coreference relation, even if uttered without any context from which references can be drawn; in this case the only BT compatible index set is the empty set, i.e. the one that does not specify any mutual dependency between the two elements. On the side of the interpretative processes, this corresponds to (possibly infinitely) many non overlapping reference assignments to the two pronouns.[12]

Anaphors make the real difference, though, since Principle A requires them to get their reference from intrasentential items. Our attention will be focused on $\mathfrak{R}_w'$, called the *A-restricted binding constraint set* and on $\mathfrak{I}_w'$, called the *A-restricted indexation set*. $\mathfrak{R}_w'$ is defined in such a way that $(\phi\ r\ \psi) \in \mathfrak{R}_w'$ iff either $\phi=\{n\}$, $n \in A$ or $\psi=\{m\}$, $m \in A$ and r is as in $\mathfrak{R}_w$. $\mathfrak{I}_w'$ is defined in a similar way. The problem, then, is to find out whether an A-reduced index set verifying (i), (ii) and (iv) of Theorem 1 exists, for a given pair (w, τ_w).

Theorem 2 - Conditions for BT Satisfiability: Let w be a sentence, τ_w one of its phrase structure tree representations and $\mathfrak{R}_w'$ its A-restricted binding constraint set; then, w satisfies BT iff for any $\phi=\{n\}$, $n \in A$ there exists an element, $\psi=\{m\}$, $m \in P, R$, such that there is a path, $p=(\phi\ r_1\ \phi_1)\ (\phi_1\ r_2\ \phi_2) \ldots (\phi_{m-1}\ r_m\ \psi)$ in $\mathfrak{R}_w'$ with string $w_p \in b^+$ and $(\psi\ d\ \phi_{m-1}) \notin \mathfrak{R}_w$.

An algorithmic solution for the satisfiability problem can be pursued by means of an approach similar to the one sketched above for verification.

5 Conclusions

BT is concerned with the relationships among the references of NPs. Indices, however, tend to collapse together situations in which more subtle distinctions seem to be needed or blur the distinctions between symmetrical relationships (coreference) and asymmetrical ones (binding).

A formalism has been provided that does not suffer the shortcomings of indexation. It permits a relevant number of phenomena to be addressed in a rather natural way and provides a richer and less ambiguous input to the semantic routines. The overall architecture can be depicted as follows: given a sentence, w, and a phrase structure tree representation τ_w, a set $\mathfrak{R}_w$ is built which, essentially, is a partial encoding of Principles A, B and C of BT as applied to τ_w. $\mathfrak{R}_w$, together with general BT compatibility conditions (see Theorem 1), constrains the form and content of any well

[12] Generalizing, it is easy to see how the empty set is a BT compatible indexation set whenever anaphors are not involved.

formed indexation set, $\Im_w$. As far as the version of BT considered here (essentially, that of Chomsky, (1986)) is concerned, the work of syntax ends with $\Im_w$; any further computation is semantic.

The formalism could be extended to other phenomena. Consider, for instance, the ban against circular reference; statements (ii) and (iii) of Theorem 1 account for the particular cases in which an item α is bound by itself or contributes to the reference of another element while being coindexed with it. More general cases were addressed by the so called i-within-i condition of Chomsky (1981) and, more recently, by the condition on circular chains of Hoeksema and Napoli (1990). The latter forbids circular chains, where a chain is a sequence of elements $a_1, ..., a_n$ such that either a_i is coindexed with a_{i+1} or a_i contains a_{i+1}. This condition could be captured within the framework proposed here by explicitly introducing dominance, say, by means of a relation symbol s_1 and, then, by requiring that no circular paths are in $\Im_w$ such that their strings are in the language $(s_1L_1)^+$. If this approach is tenable, then a parallelism emerges between the s and the s_1 relations, since both are involved in statements forbidding some kind of circularity (for s, the relevant statement is (iii) of Theorem 1) and both can be seen as estabilishing some sort of referential dependency between two items. The relevant dependency for s is set inclusion while for s_1 it is some kind of functional dependency, under the assumption that the reference of a constituent is a function of the references of its subconstituents. This observation accounts for the fact that disjoint reference constraints affect items in the s relation (see point (vi) of Theorem 1) but not those in s_1.

This work has been developed as part of a larger system that uses GB as the reference syntactic theory. Currently, we are studying two applications of the formalism presented here: 1) on-line algorithms for the satisfiability problem addressed in Section 4.2 in an off-line fashion; the interleaving of the computation of satisfiability with structure building would provide a way to rule out ungrammatical analysis of the input string at an early stage, i.e. as soon as their incapability of satisfing BT can be detected; 2) algorithms for the exhaustive generation of all index sets that are BT compatible w.r.t. a given τ_w.

ACKNOWLEDGMENTS. The author would like to acknowledge the continuous and fruitful discussions with Alessandra Giorgi and Giorgio Satta; many of the ideas in this paper have arisen during them. Of course, the responsability for any error is author's one.

REFERENCES

Aho, A.V., Hopcroft, J.E., Ullman, J.D., (1974), *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, Ma.

Barton, G., (1984), Towards a Principle Based Parser, MIT AI Memo No.788

Berwick, R., (1987), Principle-Based Parsing, MIT AI Lab Technical Rept. 972

Berwick, R., (1989), Natural Language Computational Complexity and Generative Capacity, to appear in *Computers and Artificial Intelligence*

Chomsky, N., (1981) *Lectures on Government and Binding*, Foris, Dordrecht

Chomsky, N., (1986) *Knowledge of Language*, Praeger, New York

Correa, N., (1988), A Binding Rule for Government-Binding Parsing, *Proceedings of COLING*, Budapest

Fong, S., (1990), Free Indexation: Combinatorial Analysis and a Compositional Algorithm, *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, Pittsburgh, Pe.

Giorgi, A, Pianesi, F, Satta, G., (1990), A Computational Approach to Binding Theory, *Proceedings of COLING*, Helsinki

Haïk, I., (1984), Indirect Binding, *Linguistic Inquiry*, 15, 185-224

Higginbotham, J., (1981), Reciprocal Interpretation, *Journal of Linguistic Research*, 1, 97-117.

Higginbotham, J., (1983) Logical Form, Binding and Nominals, *Linguistic Inquiry*, 14, 395-420

Higginbotham, J., (1985), On Semantics, *Linguistic Inquiry*, 16, 547-594

Hoeksema, J. and D.J. Napoli, (1990), A Condition on Circular Chains: A restatement of i-within-i, *Journal of Linguistics*, 26, 403-424.

Ingria, R.P.J. & D. Stallard, (1989), A Computational Mechanism for Pronominal Reference, *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, Vancouver.

Kolb, H.P.& C. Tiersch (1990), Levels and Empty Categories in a Principles and Parameters Approach to Parsing, in K. Netter & C. Rohrer (eds.) *Representational and Derivational Approaches to Generative Grammar*, Reidel, Dordrecht (forthcoming)

Lasnik, H., (1976), Remarks on Coreference, *Linguistic Analysis*, 2, 1-23

Lasnik, H. & J. Uriagereka (1988), *A Course in GB Syntax*, MIT Press, Cambridge, Ma.

Reinhart, T., (1983), *Anaphora and Semantic Interpretation*, University of Chicago Press, Chicago

Reinhart, T., (1987), Specifier and Operator Binding, in E.J. Reuland & A.G.B. ter Meulen (eds) *The Representation of (In)definiteness*, MIT Press, Cambridge, Ma.