# Referential Determinism and Computational Efficiency: Posting Constraints from Deep Structure*

Gavan Duffy and John C. Mallery

Department of Political Science
Massachusetts Institute of Technology
Cambridge MA 02139
Arpanet: Gavan at MIT-MC, JCMa at MIT-MC

**Abstract**

Most transformational linguists would no longer create explicit deep structures. Instead they adopt a surface-interpretive approach. We find deep structures indispensable for projection into a semantic network. In conjunction with a reference architecture based on constraint-posting, they minimize referential non-determinisms. We extend Marcus' Determinism Hypothesis to include *immediate reference*, a foundational subclass of reference. This *Referential Determinism Hypothesis*, constitutes a semantic constraint on theories of syntactic analysis, arguing for theories that minimize referential non-determinism. We show that our combination of deep structures and constraint-posting eliminates non-determinism in immediate reference. We conclude that constraint-posting, deep-structure parsers satisfy the referential determinism hypothesis.

## I Determinism

Pragmatic reasoning is necessarily non-deterministic. It is capable of non-monotonic belief-revision. It can hypothesize and then, if the hypothesis is rejected, it can backtrack, erase some or all of its structure, and start again. As more resources become available to such a reasoner, its performance improves. Its available resources are maximized when its input does not require revision and when all other components perform deterministically.

This view is consistent with Marcus' Determinism Hypothesis [23] and sympathetic to linguists' desires for parsimonious grammar specifications. Sentence analysis, of course, does not exhaust the range of interpretation necessary for sentence understanding. Reference, for example, is another crucial aspect. We offer as a corollary to Marcus' hypothesis this *Referential Determinism Hypothesis*:

> Prefer a sentence analysis that minimizes the non-determinism of reference.

We argue below that deep-structure representations with constraint-posting reference satisfy this theoretical constraint because together they minimize referential non-determinism.

## II Reference

Russell [27] first distinguished internal (private) and external (public) reference. We do not discuss external reference, neither as the correspondence of terms to an external world [26] nor as the problem of recovering the intended meanings of speakers [1, 10, 21]. Much of pragmatics also concentrates on external reference [1]. Instead, we analyze *internal* reference, wherein a hearer finds a correspondence between a sentence and his or her model of the world. We focus on the private reference of sentences using public syntax. Research conducted at SRI on the referential resolution of NPs [13, 14, 16] constitutes an important antecedent of our work. We do not, however, limit internal reference to NP resolution.

We distinguish two categories of internal reference: *immediate reference* and mediated or *deliberative reference*. Both are conceived within the constraint-posting framework discussed below. They are distinguished by their computational complexity. Immediate reference is simpler. It uses constraints that just "read off" relations from *explicit* representations in memory, creating no new structure. Immediate reference may at times utilize *spontaneous inferences*,** *e.g.*, virtual copy inheritance [3, 12]. We consider an inference spontaneous if it can be computed deterministically, creates no new structure, and requires no deliberation [8] or reflection [2].***

Deliberative reference is more complex. It may incorporate constraints requiring complex reasoning to discriminate among possible referents. Deliberative referential ability depends first on the capacity to locate the terms involved in reasoning. Immediate reference thus provides a bootstrapping foundation for deliberative reference. In our preliminary view, contradictory, incommensurate, and null references for immediate references, as well as certain ambiguous syntactic constructions, signal the need for deliberation. Its inherent complexity puts further discussion of deliberative reference beyond the scope of this paper.

## III AI and Transformational Grammar

Transformational Grammar (TG) has remained controversial in AI circles due primarily to its alleged computational intractability [cf. 30]. Berwick [4] argues that AI researchers should view modern TG not as a system of computationally intractable rules, but rather as a tractable system of constraints. Berwick advocates a surface-interpretive (SI) approach [6], in surface structure annotation guides both analytic and generative derivations. No longer are the traditional deep structure trees explicitly created. They exist only *implicitly* in the annotation. Berwick urges AI researchers to reconsider their views of TG in that light.

The transition from the traditional deep-structure approach to SI runs briefly as follows. Chomsky's "Standard Theory" [5] was open to certain criticisms. Most crucial was the observation that logical

* This research was done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-80-C-0505.

** Ken Haase [15] proposes the distinction between "spontaneous" and "deliberative" inference.

*** Since class inclusion can be computed quickly [24] and default relations need only be known to be inheritable (not actually inherited), we place our version of virtual-copy inheritance in the category of spontaneous inference. We assume that the class hierarchy does not have exceptions and that the virtual inheritances justifies nothing aside from reference.

operations can only be determined at the surface level. Jackendoff [18] suggested using deep structures for the interpretation of grammatical relations while retaining surface structures for determining quantifier scopes and other logical relations. Shortly thereafter, Fiengo [11] began development of "trace theory", in which grammatical relations are determined by the interpretation of traces (pointers) left in the surface representation. Since both the grammatical relations and logical relations could be computed from the surface level, parsimony appeared to dictate that deep structures be abandoned.

We find SI to be less parsimonious than it appears. We have found explicit deep structures indispensable for computationally efficient reference. Together with a reference architecture based on constraint posting, they eliminate needless recomputation and backtracking. Below, we examine the reference mechanism situated between the transformational Relatus parser* and the Relatus knowledge base, Gnoscere.** Below, we illustrate the efficiencies we gain by using explicit deep-structures for the analysis of grammatical relations.

### IV Immediate Reference Using Constraint Posting

A precedent for our use of constraint-posting is MUMBLE [25], which posts constraints in sentence generation rather than analysis. The general principle embodied in constraint-posting is to "wait and see" before committing to any particular course of action, thereby avoiding false starts. Backtracking is avoided because decisions are made only when full information is available.

Mapping from syntax to semantics decomposes into two recursive phases. First, the parse-tree is traversed depth-first and constraints are posted on the deep-structure nodes.*** This traversal composes a constraint tree, constructed from a subset of existing parse-tree nodes. Second, the constraint tree is traversed depth-first and nodes are referenced or created (when necessary) in some semantic network.**** Both phases sub-divide their respective tasks, localizing them to the deep-structure nodes. In the reference phase, localization makes possible the successful reference of subtrees of the constraint tree in the semantic representation even though other parts of the sentence may be new. Using the constraint tree as a match key, the reference phase implements a hierarchical match of the semantic network.

Before posting its constraints, a sentence specialist creates a canonicalized deep structure tree composed of expert nodes and headed by itself. The sentence specialist supervises constraint-posting within its scope by telling each of its immediate constituents to post their constraints, and so on, recursively. When this recursion unwinds, it leaves a constraint tree attached to the top-level VP node. The constraint tree conserves the canonical grammatical relationships explicit in deep structure. For us, these constraints constitute logical form (LF). The constraints are "public", or independent of any particular semantic network. Because they support input to a network environment suitable for logical inference, the constraints differ from the LF of TG [7].

```
(REFERENCE
  WANT
  :SUBJECT (REFERENCE
             POLICE
             :CONSTRAINTS
             ((INDIVIDUAL-P)
              (PNUMBER-OF-SUBJECT-RELATIONS HQ 1)
              (SUBJECT-RELATION HQ SECRET ((TRUE))))
             :FORCE-NEW-P NIL :PARTICULAR-P T)
  :OBJECT (REFERENCE
           ELMER
           :CONSTRAINTS ((INDIVIDUAL-P))
           :FORCE-NEW-P NIL :PARTICULAR-P T)
  :CONSTRAINTS
  ((TRUE)
   (PMSUBJECT-RELATION HAS-TENSE PAST)
   (PMSUBJECT-RELATION HAS-ASPECT PERFECT)
   (SUBJECT-RELATION
     FOR
     (REFERENCE
       CRIME
       :CONSTRAINTS
       ((INDIVIDUAL-P)
        (SUBJECT-RELATION HAS-QUANTITY PLURAL)
        (PMOBJECT-RELATION-TO-UNKNOWN
          COMMIT
          (REFERENCE-UNKNOWN *SOMETHING*
                             :CONSTRAINTS ((INDIVIDUAL-P)))
          ((TRUE)
           (PMSUBJECT-RELATION TRANSFORMED-BY PASSIVE-TRANSFORM)
           (PMSUBJECT-RELATION HAS-TENSE PAST)
           (PMSUBJECT-RELATION HAS-ASPECT PERFECT)
           (SUBJECT-RELATION
             AGAINST
             (REFERENCE
               STATE
               :CONSTRAINTS
               ((INDIVIDUAL-P)
                (PNUMBER-OF-SUBJECT-RELATIONS HQ 1)
                (SUBJECT-RELATION HQ TOTALITARIAN ((TRUE))))
               :FORCE-NEW-P NIL :PARTICULAR-P T)))))
       :FORCE-NEW-P NIL :PARTICULAR-P T))))
```

**Figure 1:** Constraint tree for *"The secret police wanted Elmer for some crimes that were committed by the totalitarian state."*

The constraint tree for the sentence "The secret police wanted Elmer for some crimes that were committed against the totalitarian state" is presented in Figure 1. Note that every appearance of the symbol *reference* marks points where a single reference takes place, and thus the points of semantic composition. The symbol following *reference* is the token type to be referenced. The keyword *:constraints* is followed by the list of constraints applied in the reference. The keywords *:subject* and/or *:object* signify that a relation is being referenced. They designate the subject and object of that relation (objects are omitted for unergative verbs). So in Figure 1, the subject of the 'want' relation is the 'police' reference and the object is the reference to an 'Elmer'.

In the reference to 'police', the *individual-p* constraint means we are looking for an individual 'police' as opposed to its universal (the class of 'police'). The *subject-relation* constraint means that we want a 'police' which participates as the subject of an HQ (has-quality) relation to 'secret'. A subject-relation constraint is an *ordinary constraint*. Ordinary

* The Relatus parser is a deterministic, transformational parser which creates deep syntactic structure trees composed of intelligent nodes, was implemented by Duffy [9]. Credit for demonstrating the computational feasibility of a real-time transformational parse should go to Katz [20].

** The knowledge base is a semantic network that provides a constraint-posting reference mechanism and a set of reference contraints. It was implemented by Mallery. Gnoscere is built out of a frame system and implements multiple semantic networks, one for each represented belief system. These networks are relational. Winston [31] showed uses of a relational semantic representation in analogical reasoning. The mechanism that converts syntactic deep structure trees into constraint trees suited to Gnoscere's reference system was implemented by Mallery in close collaboration with Duffy. Research on the Relatus system is still preliminary.

*** A parallel algorithm would fan out top-down according to the branching factor of the deep structure.

**** A parallel algorithm would perform reference of the constraint tree bottom-up from the bottom-most non-terminal nodes, "fanning into" the sentence at the top.

constraints are necessary. A successful referent must satisfy them. The inverse of a subject-relation constraint is an *object-relation* constraint. Only relations have truth value constraints (e.g. *true*). Relations and objects may both have quantificational constraints (e.g., *universal-p*).

A *P* prefix distinguishes *preference constraints* from ordinary constraints. Successful referents need not satisfy preference constraints. These order successful candidates by unweighted voting, aiding selection of the most promising candidate for reference. *Pnumber-of-subject-relations*, for example, finds a referent preferring one HQ relation over others. *Preferred-mandatory constraints*, prefixed with *PM*, also use the voting scheme to order the referential possibility space, but they require the successful referent to have the feature they specify. For example, if the referent for the 'want' relation in the top-level of Figure 1 does not have <subject-relation has-tense past>, this relation is created for it.

Constraints are posted directly on the deep-structure nodes. They may be displaced from lower nodes to a higher node. When a node's constraints are raised onto the constraints of another, the displaced node does not itself appear in the resulting constraint tree. It is represented instead in the raised constraints. An example of displacement occurs in Figure 1 where the 'commit' relation uses the *pmobject-relation-to-unknown* constraint to restrict the reference for 'crime', requiring it to possess a certain object relation with an unspecific subject. Constraint displacement occurs for all adverbs, relative clauses, and prepositional phrases. Deep structure, by virtue of its hierarchical connectivity, explicitly encodes a unique, grammatically canonical plan for constraint-tree construction.

Parse-tree nodes exploit their internal structure to post reference constraints. Constraint posting is simplified by the presence of a deep-structure tree because all positional decisions, including embeddedness, are pre-determined. Restrictive relative clauses, for example, appear in deep structure as adjectival modifiers of an NP. The reference constraints for "The man who was wanted by the police laughed", presented in Figure 2, contain a restrictive relative on 'man' (<PMobject-relation want . . .>). Since the parse tree includes an explicit transform of the passive embedded S, the transform's constraints are easily percolated to 'man'. Because the transformation is already represented explicitly, there is no need to "unpack" the embedded S. The constraints then select the appropriate 'want' referent (created previously by the constraint tree in Figure 1) from the network.

```
(REFERENCE
  LAUGH
  :SUBJECT (REFERENCE
        MAN
        :CONSTRAINTS
        ((INDIVIDUAL-P)
         (PMOBJECT-RELATION
           WANT
           (REFERENCE
             POLICE
             :CONSTRAINTS ((INDIVIDUAL-P))
             :FORCE-NEW-P NIL :PARTICULAR-P T)
           ((TRUE)
            (PMSUBJECT-RELATION TRANSFORMED-BY PASSIVE-TRANSFORM)
            (PMSUBJECT-RELATION HAS-TENSE PAST)
            (PMSUBJECT-RELATION HAS-ASPECT PERFECT))))
                    :FORCE-NEW-P NIL :PARTICULAR-P T)
  :CONSTRAINTS ((TRUE)
                (PMSUBJECT-RELATION HAS-TENSE PAST)
                (PMSUBJECT-RELATION HAS-ASPECT PERFECT)))
```

**Figure 2:** Constraint tree for *"The man who was wanted by the police laughed."*

Relative clauses exemplify local decision-making by an NP specialist. If the NP scopes a clause, it tells the clause to pass back its constraints and posts them as referential restrictions on the noun. Local decision-making also facilitates implementation of inter-constituent syntactic constraints, *e.g.*, the interconstraints between the subject and object (predicate nominal) of the verb "to be". Whenever a "be" VP is told to post its constraints, it knows to classify its subject and object as individuals or universals according to the definiteness or classness of each [19, 22]. It then adds this as constraints on the subject and object.

Once the constraint tree is complete the reference phase begins. In immediate reference we must always find any referent indexed by the symbol and its associated constraints. Only when no node in the semantic network satisfies the referential constraints is a new node created. Any other behavior would constitute a reference failure, requiring subsequent backtracking.

Sentence constituents are referenced in a semantic network bottom-up, working up the constraint tree to the sentence at the top. Thus, any extant semantic objects corresponding to constituents at any level in the trees may successfully be referenced.

Each non-terminal deep-structure node references itself, using the symbol of its head and its associated constraints. The reference mechanism of the semantic representation finds an existing semantic object which satisfies its constraints. If no suitable node is found, a new one is created according to the symbol and constraints. Once a referent is found (or created), it is returned to the superior constraint tree node. The superior object then references itself using the returned referent in the position designated by the constraint tree, and in turn, returns its own reference. The structure of the constraint tree determines the position of referents and order of reference for each node.

## V Determinism of Immediate Reference

Minimally, an SI approach would require recomputation of intermediate data structures for each interpretive "access" of a parse object. Quite possibly, an SI approach would be forced to backtrack in the composition of constraint trees. A third alternative would not canonicalize the syntax, forcing the reference and reasoning components to unpack the non-canonical structures that are passed forward. Whichever option is chosen, the SI approach would clearly be less elegant and less efficient than the deep-structure approach. It would add considerable overhead in determining (often repeatedly) positional and order-of-evaluation relationships between the various parse-objects. In contrast, the deep structure approach caches this information explicitly. The information is simply read off the tree without repeatedly "unpacking" the information.

Constraint posting from deep structure is the critical factor that makes immediate reference deterministic. No action is taken without full information. Use of full information precludes any need to retract premature references. False starts can therefore only occur in deliberative reference. Canonical parse trees simplify constraint-posting by constituents because the constituents need only enough knowledge to handle the canonical case. There is no need to recompute constraints or store them in *ad hoc* structures because parse tree nodes remember their constraints. The deep structures localize and simplify the algorithms for constraint composition and reference.

The transformation of sentences and constituents to canonical form before reference has three major beneficial effects. First, it *guarantees* the (syntactically) canonical nature of the semantic network and thus minimizes semantic backtracking (belief revision). The examples in the figures above illustrate this point.* Second, the resulting canonical semantic representation further simplifies reference. Since both the constraint tree and the representation are syntactically canonical, reference will not fail due to syntactic irregularities. Third, canonicalization of the semantic representation eliminates any need for syntactic transformations for all operations on the representation requiring reference (*e.g.*, learning). The parser effectively compiles out

syntax, making the semantic representation more efficient.

Consider the constraints presented in Figure 3 for the NP "the wanted man" in the sentence "Elmer knew then that he was the wanted man". The participial adjective, 'wanted', has been expanded into an embedded S, with the proviso that the subject of the 'want' relation is unknown. The constraint mechanism successfully discovers that this 'want' relation is the same 'want' relation as that of Figures 1 and 2.* If, as in [4], participial adjectives and participles in what are ordinarily termed passive constructs (*e.g.*, Figure 2) were treated as one-place properties just as any adjective is treated, measures to control for syntactic variation would be required, and consequently, a significant inefficiency would result. For every reference, each property would need to be checked to see whether it has previously been expressed as a relation, and *vice versa*. The availability of effective resources would be further reduced by a factor which is a function of the size of the semantic units manipulated. The cost becomes steeper as application size increases and must be paid repeatedly -- on each reference.

```
(REFERENCE
  KNOW
  :SUBJECT (REFERENCE
            ELMER
            :CONSTRAINTS ((INDIVIDUAL-P))
            :FORCE-NEW-P NIL :PARTICULAR-P T)
  :OBJECT
  (REFERENCE
    BE
    :SUBJECT (REFERENCE
              ELMER
              :CONSTRAINTS ((INDIVIDUAL-P))
              :FORCE-NEW-P NIL :PARTICULAR-P T)
    :OBJECT (REFERENCE
             MAN
             :CONSTRAINTS
             ((INDIVIDUAL-P)
              (PMOBJECT-RELATION-TO-UNKNOWN
                WANT
                (REFERENCE-UNKNOWN
                  *SOMETHING*
                  :CONSTRAINTS ((INDIVIDUAL-P)))
                ((TRUE)
                 (PMSUBJECT-RELATION HAS-TENSE PAST)
                 (PMSUBJECT-RELATION HAS-ASPECT PERFECT))))
             :FORCE-NEW-P NIL :PARTICULAR-P T)
    :CONSTRAINTS ((TRUE)
                  (PMSUBJECT-RELATION HAS-TENSE PAST)
                  (PMSUBJECT-RELATION HAS-ASPECT PERFECT)))
  :CONSTRAINTS ((TRUE)
                (PMSUBJECT-RELATION HAS-TENSE PAST)
                (PMSUBJECT-RELATION HAS-ASPECT PERFECT)
                (SUBJECT-RELATION HQ THEN NIL)))
```

**Figure 3:** Constraint tree for *"Elmer knew then that he was the wanted man."*

## VII Conclusions

We have illustrated how deep structures combine with constraint-posting reference to improve the efficiency of immediate reference, a central aspect of any interface between syntax and semantics. Canonicalization of syntax makes immediate reference deterministic, and thus increases the efficiency of all operations which build on immediate reference, including deliberative reference, reasoning, and learning.

Constraint-posting reference is not impossible, in principle, within the context of an SI approach to sentence analysis. From a syntactic standpoint, our approach and the SI approach are mere notational variants [7]. Only when mapping from syntax to semantics do the full set of computations relevant to parsimony claims emerge. At the very least, an SI parser with constraint-posting reference would be forced to repeatedly recompute its relationships to deep-structure, or be forced to backtrack in the computation of constraints. Alternatively, it may simply export its non-canonicalities to the semantic component. Because this alternative would force additional chores on the semantic component and thus slow all reasoning activities, our objection to it is concomitantly more strenuous.

Canonicalization of its input enhances consistency in the semantic component. Explicit representation of both surface and deep structures enhances the elegance and efficiency of the composition of referential constraints. These constraints, in turn, maximize the determinism of reference, thereby minimizing the amount of non-deterministic backtracking (belief-revision) necessary in the semantic component. The structure of the referential constraints inherit the canonicalized grammatical relations of deep structure. Inference operations in the semantic network can rely on that canonicalization and thus perform better. A variant of SI which supports a canonical semantic representation may someday be invented, but the parsimony argument for SI would no longer remain tenable.


### Acknowledgements

This paper was improved by comments from Steve Bagley, John Batali, Bob Berwick, Carl Hewitt, Joel Isaacson, and Rick Lathrop. Margaret Fleck provided valuable comments. Robert Ingria helped refine our arguments with detailed comments. Responsibility for the content, of course, remains with the authors.


### References


[1] Bach, K. and R. M. Harnish, *Linguistic Communication and Speech Acts*. Cambridge, Mass.: MIT Press, 1979.

[2] Batali, J. "Computational Introspection." Memo No. 701, MIT Artificial Intelligence Laboratory, Cambridge, Mass.. February 1983.

[3] Bawden, A., D. Hillis, and D. A. McAllester, "Virtual Copy Compression." seminar, MIT Artificial Intelligence Laboratory, Spring, 1982.

[4] Berwick, R. C., "Transformational Grammar and Artificial Intelligence: A Contemporary View." *Cognition and Brain Theory*. 6:4 (1983) 383-416.

[5] Chomsky, N., *Aspects of the Theory of Syntax*. Cambridge, Mass.: MIT Press, 1965.

[6] Chomsky, N., *Lectures on Government and Binding*. Dordrecht: Foris, 1981.

[7] Chomsky, N. "On the Representation of Form and Function." *The Linguistic Review* 1:1 (1981-82) 3-40.

[8] Doyle, J., *A Model for Deliberation, Action, and Introspection*. Doctoral dissertation. MIT. May 1980.

[9] Duffy, G., "Parsing with Smart Nodes." forthcoming. 1984.

[10] Evans, G., *The Varieties of Reference*.. New York: Oxford University Press, 1982.

[11] Fiengo, R. W., *Semantic Conditions on Surface Structure*. Doctoral dissertation. MIT. 1974.


* Some oppose the derivation of such adjectival participials from sentential sources as involving *ad hoc* rules. such as "Whiz" deletion [*cf.* 29] However, Ingria [17] argues that this analysis may be restated without resort to *ad hoc* processes.

[12] Fahlman, S., *NETL: A System for Representing and Using Real-World Knowledge.*, Cambridge, Mass.: MIT Press, 1979.

[13] Grosz, B. J., "Discourse Analysis." in [28]. 235-268.

[14] Grosz, B. J., "Resolving Definite Noun Phrases." in [28]. 287-298.

[15] Haase, K., *ARLO: The Implementation of a Language for Describing Representation Languages.* Bachelor's Thesis. MIT. 1984.

[16] Hendrix, G. G., "The Representation of Semantic Knowledge." in [28]. 121-181.

[17] Ingria, R. J. P., "Rehabilitating a Classical Analysis: Clausal Sources for Nominal Modifiers." forthcoming. 1984.

[18] Jackendoff, R. S., *Semantic Interpretation in Generative Grammar.* Cambridge, Mass.: MIT Press. 1972.

[19] Jackendoff, R. S., *Semantics and Cognition.* Cambridge, Mass.: MIT Press. 1983.

[20] Katz, B., and P. H. Winston "Parsing and Generating English Using Commutative Transformations." Memo No. 677. MIT Artificial Intelligence Laboratory. Cambridge, Mass. May 1982.

[21] Lyons, J., *Semantics.*, Vol. 1, Cambridge, U.K.: Cambridge University Press. 1977.

[22] Mallery, J. C., "Universality and Individuality: The Interaction of Noun Phrase Determiners in The Case of The Verb 'To Be'." forthcoming. 1984.

[23] Marcus, M. P., *A Theory of Syntactic Recognition for Natural Language.* Cambridge, Mass.: MIT Press. 1980.

[24] McAllester, D. A., "An Outlook on Truth Maintenance." Memo No. 551. MIT Artificial Intelligence Laboratory. Cambridge, Mass. August, 1980.

[25] McDonald, D. M., "Natural Language Generation as a Computational Problem: An Introduction." in R. C. Berwick and M. Brady, *Computational Models of Discourse.* Cambridge, Mass.: MIT Press, 1983, pp. 208-265.

[26] Putnam, H., *Reason Truth and History.* Cambridge, U.K.: Cambridge University Press, 1981.

[27] Russell, Bertrand, *Human Knowledge: Its Scope and Limits.* New York: Simon and Schuster, 1948.

[28] Walker, D. E., ed., *Understanding Spoken Language.* New York: North-Holland, 1978.

[29] Williams, E. S., "Small Clauses in English." in J. Kimball, ed., *Syntax and Semantics/* Vol 4., New York: Academic Press, 1974.

[30] Winograd, T., *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language.* Doctoral dissertation. MIT. February 1971.

[31] Winston, P. H., "Learning New Principles From Precedents And Exercises." *Artificial Intelligence* 19:3 (1982).