Massachusetts Institute of Technology
The Artificial Intelligence Laboratory

AI Working Paper 274 June 1985

Toward a Principle-Based Translator

by
Bonnie J. Dorr

Abstract

A principle-based computational model of natural language translation consists of two components: (1) a module which makes use of a set of principles and parameters to transform the source language into an annotated surface form that can be easily converted into a "base" syntactic structure; and (2) a module which makes use of the same set of principles, but a different set of parameter values, to transform the "base" syntactic structure into the target language surface structure. This proposed scheme of language translation is an improvement over existing schemes since it is based on interactions between principles and parameters rather than on complex interactions between language-specific rules as found in older schemes.

The background for research of the problem includes: an examination of existing schemes of computerized language translation and an analysis of their shortcomings. Construction of the proposed scheme requires a preliminary investigation of the common "universal" principles and parametric variations across different languages within the framework of current linguistic theory.

The work to be done includes: construction of a module which uses linguistic principles and source language parameter values to parse and output the corresponding annotated surface structures of source language sentences; creation of procedures which handle the transformation of an annotated surface structure into a "base" syntactic structure; and development of a special purpose generation scheme which converts a "base" syntactic structure into a surface form in the target language.

1 Introduction

This proposal is divided into 6 sections. The second section describes the background for research of the problem of computerized natural language translation. Existing translation schemes will be discussed and the shortcomings of these schemes will be addressed. Also, the reasons that a principle-based scheme would be an improvement over existing schemes will be briefly outlined.

A plan for the development of a theoretically based computational scheme will be introduced in the third section. The manner in which the proposed scheme embodies linguistic theory will be explained via a presentation of examples that illustrate how commonalities and variations between Spanish and English can be mapped to principles and parameters of modern linguistic theory. Also, the goals of the proposed scheme will be described.

The fourth section presents a description of the work that needs to be done. There are several possibilities for the implementation of the proposed scheme. One possible approach is a modification of the Marcus parser so that it operates on a constrained grammatical theory of principles and parameters. This section consists of a description of the modifications that would be required in order to accommodate the proposed scheme. Also, a discussion of some of the difficulties that might arise in the development of the proposed scheme is included.

Resources for the proposed research are enumerated in the fifth section. The sixth section provides a listing of some of the references the research will require.

2 Background for Research

2.1 Early Machine Translation Designs

Over the last 35 years, several approaches to machine translation have been taken. Early theories of machine translation brought forth "direct" and "local" designs. The transition from source to target consisted in a word-for-word replacement, followed by a limited amount of transposition of words to result in something vaguely resembling English. A system which used such a design was the Georgetown Automatic Translation (GAT) system which began in 1952.[1] There was no true linguistic theory underlying such a design.

2.2 Transformational Grammar Approach to Machine Translation

As Chomsky's transformational paradigm quickly gained popularity in the 1960's, machine translation systems became more oriented toward a syntactic interlingua based on deep structures. The Mechanical Translation and Analysis of Languages (METAL) system at the Linguistics Research Center at the University of Texas began its development in 1961 within this framework.[2] This system employs a phrase-structure grammar that is lexically controlled. Rule application is restricted via constraint-enforcement procedures associated with the context-free rules. To deal with ambiguity, the system makes use of plausibility factors to select a "best" interpretation from ambiguous sentences. Furthermore, a transformational component which indexes transformations to specific syntax rules is required.

[1] This is discussed in Slocum, May 1984.

[2] Slocum, April 1984.

Slocum argues that certain compromises had to be made since transformational linguistics was not sufficiently well-developed to support an operational system. One compromise made during the METAL development was the introduction of a "transfer" component which maps "shallow analyses of sentences" in the source language into "shallow analyses of equivalent sentences" in the target language. Thus, the "deep representation" framework that originally had been the basis of the project's development was abandoned.

The METAL system is afflicted with several maladies. First of all, the system in its entirety is enormous since it has several components. Secondly, even before translation begins, the text must pass through several preprocessing stages: first the text enters an annotation component (the output of which requires human verification and emendation); then it must pass through translation preparation module; and finally, it enters a dictionary pre-analysis stage.

In addition to the size and complexity of METAL, a third and fundamental problem of METAL is that the the rules employed by the system are detailed and language-specific. For example, a simple German context-free phrase structure rule for building a noun stem and an inflectional ending into a noun appears as follows:

```
NN              NST             N-FLEX
0               1               2
(LVL 0)         (REQ WI)        (REQ WF)

TEST      (INT 1 CL 2 CL)

CONSTR    (CPX 1 ALO CL)
          (CPY 2 NU CA)
          (CPY 1 WI)

ENGLISH   (XFR 1)
          (ADF 1 ON)
          (CPY 1 MC DR)
```

Although this rule is equivalent to the simple context-free rule NN ⇒ NST N-FLEX, it contains several complex parts: a constituent test that checks the sons to ensure their utility in the current rule; an agreement TEST to enforce syntactic correspondence among constituents; a phrase CONSTRuctor which formulates the interpretation defined by the current rule; and one or more target-specific transfer rules. Furthermore, these components may include calls to case frame or transformational procedures as well as simpler routines to test and set syntactic and semantic feature values.

The LRC MT system is currently equipped with approximately 550 rules and 10,000 lexical entries in each of the two main languages (German and English). The complexity and language-specific nature of the rules translate into several problems. First of all, because the rules and lexical entries are so complex, the subject area must be very limited. Secondly, each rule is highly language-dependent in character; thus, there must be a set of target-specific transfer rules for every language that will serve as a target. This means that the rule system grows rapidly as each target language is added to the system. Thirdly, the rules

are very stipulatory since there are no theoretical reasons for the rules being the way they are. Finally, each rule must carefully spell out the details of its application; thus, there is no way to capture linguistic generality among the rules in the system since general constraints are not factored out of the syntactic rules.

2.3 Semantic-Based Approach to Machine Translation

At the other end of the spectrum of machine translation are those systems which reject syntax as a basis of language translation. Rather, translation is treated almost entirely on the basis of semantics, guided by a strong underlying model of the current situational context and expectations. An example of a system which operates within this framework is the MOPTRANS (MOP-based TRANSlator)[3] system which relies on a hierarchically organized knowledge representation called MOPs (Memory Organization Packets).[4] The system, which is currently being implemented at Yale, uses a semantic-based method of interpretation to translate news stories about terrorism and crime from Spanish to English.

Lytinen claims that there are two problems with syntactic-based translation systems that motivate an entirely semantic-based design. The first is the number of rules the system must deal with. A word may have several word senses, and each word sense would require a myriad of rules specifying the contexts in which the word sense might appear. A second problem is an indexing problem. Since there are thousands of rules to choose from, "the amount of information the system would have to look for would be enormous, and deciding what information in the sentence was relevant for disambiguating the word in each particular context would be impossible."[5]

The two types of semantic knowledge used in MOPTRANS are: *abstraction knowledge*, which is a set of general scenes containing similar elements shared among specific contexts; and *packaging knowledge*, which is a set of episodes containing sequentially ordered events that are likely to occur together in the world. The primary idea behind MOPTRANS is that knowledge which is common to many different situations is stored in one processing structure, and this processing structure is available for all situations to which it applies.

The problem with this semantic-based system is that it is requires massive amounts of storage and computation time. The two basic components of MOPTRANS are the parser and the generator. The generator is not well-defined in any of the MOPTRANS papers by Lytinen and Schank. However, in order to get an inkling for how complicated the system is, one only needs examine the parser since it is a complex enough system within itself. The parser consists of a *dynamic processing* component which generates MOPs on the fly as they are needed. In addition, it contains *prediction* and *inferencing* modules which in turn execute *specialization* and *instantiation* routines. Furthermore, there are several background components used by the MOPTRANS parser. The primary background elements are MOP *scenes*, which are sequentially ordered events; and MOP *abstractions*, which are hierarchically organized structures that relate similar MOP scenes. An additional background component for the MOPTRANS system is a semantically based *dictionary*. A

[3]Lytinen and Schank, 1982.

[4]Schank, 1979.

[5]Lytinen and Schank, p. 13, 1982.

final component of the parser is a set of *general rules* which control the parsing process.[6]

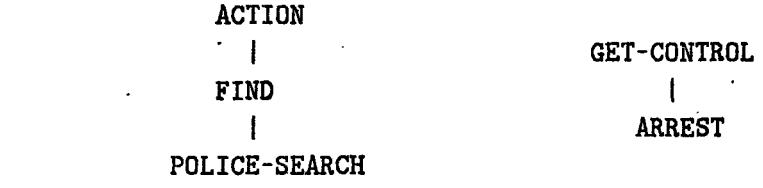
The complexity of the MOPTRANS parser is illustrated by the translation of the sentence:

Spanish:
La policia *realiza intensas diligencias* para capturar a un presunto maniatico sexual que dio muerte a golpes y a puñaladas a una mujer de 55 años.

English:
Police *are searching for* a presumed sex maniac who beat and stabbed to death a 55-year-old woman.

The phrase *realiza intensas diligencias* is a very general phrase which can be translated in many different ways depending on the context in which it appears. Literally, *realiza intensas diligencias* means *to realize diligent actions.* However, some of the other English translations of the phrase are: *to run errands, to search for, to shop for, to go* and *to determine.* The following abstractions and scenes are required to translated the sentence unambiguously:

Abstractions:



Scenes:

MGET = KNOW + FIND + GET-CONTROL

M-POLICE-CAPTURE = POLICE-INVESTIGATION + POLICE-SEARCH + ARREST

In the dictionary, *diligencias* and *capturar* are simply defined as ACTION and GET-CONTROL respectively.

[6] A more thorough overview of the parsing components and their interactions is presented in Dorr, 1983.

The following *general rules* are required to control the parsing process:

```
RULE 1: GENERAL SCENE EXPECTATION DISAMBIGUATION RULE
 If a MOP is active which predicts that a scene S will occur, and if a
 word in the story is encountered which refers to an abstraction of the
 scene S, then that word should be disambiguated to mean S, and S
 should be instantiated.

RULE 2: MOP LOCATION INSTANTIATION RULE
 If a setting or location is mentioned which is associated with a
 particular MOP, and a person who would be likely to take part in that
 MOP is mentioned, then instantiate the MOP.

RULE 3: ACTOR LOCATION INSTANTIATION RULE
 If a person is in a location in which HE typically engages in a
 particular MOP, then instantiate that MOP.

RULE 4: INSTANTIATION PRECEDENCE RULE
 If both rule 2 and rule 3 apply in a story, use only rule 3.
```

Through a very complex process of prediction, specialization and instantiation (the details of which are discussed in the literature), the parser finally arrives at a semantic representation which is processed by the generator and converted into the following translation:

The police are searching for a sex maniac because he killed a 55 year old woman.

The claim that rule-based syntactic systems are both too large and too complex to adequately handle natural language translation may be well-grounded, but this semantic-based approach certainly does not combat the problem! In attempting to tackle the problem of word disambiguation the system incorporates an incredibly massive amount of knowledge, at the expense of providing access to a very limited domain of subject matter. Furthermore, the MOPTRANS parser not only requires scenes, abstractions and general rules, but it also requires additional information which is not mentioned in the literature. For example, rules which are specific to certain MOPs must be stored in the system so that slot-filling predictions will be correct (*e.g.*, the system must somehow figure out that POLICE performing a FIND are performing a POLICE-SEARCH in order to instantiate the appropriate POLICE-SEARCH MOP).

An additional drawback to this semantic-based approach is that there is a loss of structure and style in the final translation. Although the *deep contextual meaning* of the input text is preserved, there is a loss of emotional impact. In the POLICE-SEARCH example above, the adjectives *presumed* and *intense* are lost and the verbs *beat* and *stabbed* have been changed to the single verb *kill*. Lytinen claims that any other system which attempts to preserve structure and style without the knowledge necessary for text understanding would often produce unreliable translations. However, it is not clear that loss of emotional content does not constitute loss of the full meaning of the text. Most likely, the speaker chooses certain adjectives and verbs to convey an important *feeling* about the situation. The absence

of the vehicle of emotional conveyance might result in a complete misinterpretation of the text.

Recall that the parser only contributes toward *part* of the overall complexity of MOP-TRANS system; the generator component (which is not discussed in the literature) introduces additional complexity to the system.

2.4 Hope for the Future: Principle-Based Approach to Machine Translation

The rule systems that are the basis of existing natural language translators are still large, detailed and complicated. If the basis of machine translation design is shifted from complex, language-specific rules systems to modular theories of syntax consisting of systems of principles and parameters, several of the problems associated with earlier theories will be solved. Grammars will no longer be huge and complicated; small sets of principles will replace complicated non-explanatory designs; and strong constraints can be placed on a small basic description of an individual languages.

3 Proposed Scheme

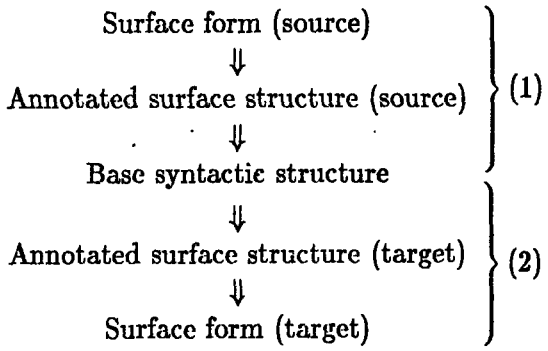
3.1 Two Step Translation Process

A Machine translation scheme based on systems of principles and parameters consists of a two step process:

1. The input sentence in the source language must be transformed into an annotated surface structure (a bracketed form which contains traces of moved constituents) which can then be converted into a "base" syntactic structure.

2. The "base" syntactic structure must then be transformed into the annotated surface structure of the target language. The surface form is then retrieved simply by removing the annotations.

Thus, the overall picture is as follows:



Note that this strategy is based on the assumption that the base syntactic structure is a form that is common to the two languages.

The syntactic structure of the input sentence must be recovered via a parser which builds an explicit tree-like structure (the annotated surface structure). However, we would not want the parser to build this structure on the basis of a set of language-dependent rules. Rather, we would want the parser to operate on the basis of a set of values for parameters associated with the input language. Thus, when the conversion into the target language takes place, the principles upon which the parser operates will remain the same, but the values of the parameters are subject to variation.

Transformation into the base syntactic structure consists of applying operations which drop moved constituents back into the place from which they were moved. Because the parser knows the values of the parameters in the source language, it knows how far moved constituents have traveled and it can also recover deleted elements. One problem that could arise during the transformation into this "common" base form is that the information about moved constituents (*i.e.*, the traces) has been lost. Thus, the parser must provide some way of annotating the base structure so that it reflects the fact that movement has taken place during this conversion.

Conversion to the source language consists of applying an "inverse" parsing strategy: if movement has taken place in the source language, the corresponding movement must take place in the transformation to the target language. Of course, this time the values for the parameters upon which the "inverse" parser is operating are based upon the target language, not the source language.

3.2 Embodiment of Linguistic Theory

The above scheme of machine translation should be constructed in such a way that properties which are shared among all languages are handled by a unified set of "core" linguistic principles, while the differences among languages are accounted for by a set of possible parameters of variation. According to Chomsky, many properties of particular languages can be accounted for through the interaction of principle-based subsystems, while complexes of properties differentiating otherwise similar languages should (ideally) be reducible to a single parameter, fixed in one or another way.[7] Thus, in order to build a machine translation system, it is necessary to determine both the properties that are universal across languages, as well as the parametric variations between languages. This requires construction of a catalogue of detailed descriptions of particular languages (*i.e.*, several examples of human translation between languages) and an abstraction of common properties from those detailed descriptions.

I will first present some examples of (human) translations from Spanish to English. I will then explain how several properties that are common to both Spanish and English can be abstracted from these examples and mapped to principles that are part of the modern transformational theory called *government-binding theory* or GB-theory. Once the commonalities are abstracted from these examples, I will then discuss how the observed differences can be mapped to parameters whose values are free to vary from language to language.

[7] A brief overview of the principles of GB-theory is presented in Barton, 1984.

3.2.1 Translation Examples

The examples presented here are from Esther Torrego's paper on Spanish inversion.[8] The first type of inversion that she discusses is "free subject inversion." The assumption is that free subject inversion moves the NP subject to the right, adjoining it to the VP:

(i) Contestó la pregunta *Juan*
'Answered the question John.'

Note that there is no such inversion rule available in English. In particular, Subject-Aux inversion (SAI) in English cannot be considered an equivalent inversion rule since the former occurs only in +Wh matrix clauses, whereas the latter is always applicable regardless of whether or not the clause is matrix, embedded, +Wh or -Wh. Furthermore, SAI is an obligatory rule, whereas free subject inversion is optional.

The second type of inversion rule presented by Torrego is "Verb Preposing" (V-Preposing). While "free subject inversion" is an optional rule which is applicable under any circumstances, V-Preposing is an obligatory rule which occurs only in clauses in which Wh-movement takes place:

(ii)

(a) Con quién vendrá *Juan* hoy?
'With whom will John come today?'

(b) *Con quién *Juan* vendrá hoy?

Just as for "free subject inversion", V-Preposing is not equivalent to English SAI. The main difference between V-Preposing and SAI is that V-Preposing can occur *in both embedded and nonembedded clauses*, whereas SAI (and also Verb/Second in Germanic languages) is restricted to root sentences. The following sentences illustrate this point:

(iii)

(a) Qué pensaba *Juan* que le había dicho *Pedro* que había publicado *la revista*?
'What did John think that Peter had told him that the journal had published?'

(b) *Qué *Juan* pensaba que *Pedro* le había dicho que *la revista* había publicado?

An even more fundamental difference between SAI and V-Preposing is that SAI requires there to be an auxiliary verb in order for inversion to take place; by contrast, in Spanish, a projection of V (not Aux) is moved to the left of the subject.[9] In fact, in Spanish, there is no syntactic constituent that is equivalent to the auxiliary verb. Thus, if there is a verbal sequence involving more than one verb in Spanish, the V-Preposing that occurs after Wh-movement does not map directly to the SAI rule application in the equivalent English sentence:

(iv) Con quién podrá *Juan* ir a Nueva York?
'With whom will John be able to go to New York?'

[8] Torrego, 1984.

[9] As described by Torrego, V-Preposing moves a V projection out of VP, adjoining it to the right of COMP under a new S node.

Furthermore, in Spanish, not all verb sequences are allowed to be split into two pieces (as in the *poder* + *ir* case above). Certain Spanish verb sequences must be moved as unbroken units during the application of V-Preposing. For example, verb sequences such as *haber* + *past-participle* and *ser* + *adjectival-phrase* must remain unbroken during V-Preposing:

(v)

(a) Qué ha organizado *la gente*?
'What have the people organized?'

(b) *Qué ha *la gente* organizado?

Finally, V-Preposing differs from SAI in that not every occurrence of a Wh-word in Comp position in Spanish causes V-Preposing; by contrast, English SAI is obligatory regardless of what the Wh-word is:

(vi) Cuándo *Juan* consiguió por fin abrir la puerta ayer?
'When did John finally get to open the door yesterday?'

Wh-phrases that do not require inversion include *en qué medida* 'in what way', *por qué* 'why', *cuándo* 'when', and *cómo* 'how'. Torrego refers to a word that makes inversion obligatory as a *Wh-word*$_A$. Thus, the inversion rule is stated as follows:

> In Spanish, a *Wh-word*$_A$ in the Comp position of a tensed clause triggers obligatory inversion in both main and embedded clauses.

3.2.2 Principles and Parameters

The translation examples above provide strong evidence for a successive cyclic analysis of Wh-movement in Spanish. Under the successive cyclic analysis proposed by Chomsky,[10] Wh-movement is free to move a Wh-phrase further than one bounding node to the left only if it moves from Comp to Comp on successive cycles. This principle could be considered a language "universal" if we allow the parameter of variation to be the bounding node. Torrego argues that V-Preposing proves to be relevant to determining the choice of bounding nodes for Subjacency in Spanish. The data presented in certain examples of Spanish sentences bear out the possibility that S is not a bounding node in Spanish in English as it is in Spanish:[11]

(vii) En qué vía dijo *Juan* que anunció el altavoz que *el tren* se estacionaría?
'What track did John say that the loudspeaker announced that the train would arrive at?'

Note that no V-Preposing occurs in the embedded clause "*el tren* se estacionaría." This is because movement of a *Wh-phrase*$_A$ is allowed to skip one cycle since S is not a bounding node.

[10] Chomsky, 1977.

[11] According to Rizzi, 1978, S is not a bounding node in Italian. However, in order to show this, Rizzi makes an appeal to the fact that Italian allows Wh-Movement out of a clause introduced by a Wh-phrase. The reason this result cannot be achieved via an appeal to V-Preposing is that V-Preposing occurs only in matrix clauses in Italian.

In addition to the principle of Subjacency and the parameter of bounding node, Torrego's examples of Spanish inversion also illustrate that V-Preposing accounts for the the principle of empty categories (ECP) and the parameters under which it operates. Since ECP accounts for a wide range of phenomena,[12] it could be considered a language "universal" just as is Subjacency. The parameter of variation that is illustrated in the examples given by Torrego is based on the observation that that while object extraction cannot lead to ECP violations in English, extraction of objects and other verbal complements in Spanish may cause ECP violations. The following example shows such a case:

(viii)

(a) Qué dices que no te explicas por qué *Juan* se habrá comprado?
'What do you say that you don't understand why John will have bought?'

(b) *Qué dices que no te explicas a quién (le) habrá comprado *Juan*?
'What do you say that you don't understand for whom John has bought?'

According to Torrego, the above phenomena occur because once the verb is preposed, it no longer governs a trace that is in the VP. Rather, it properly governs the subject position to its right.

In addition to this parametric variation associated with ECP, an additional parameter of variation is introduced by Torrego. The notion of "proper government" which determines whether or not ECP is violated must be parameterized so that it allows a trace to be properly governed when it is part of a *chain*, all of whose elements are properly governed.[13]

3.3 Goals of the Proposed Scheme

If the proposed scheme is to handle the examples mentioned above, it should be based on interacting principles and parameters rather than on rules that individually determine the details of their operation (as in the earlier schemes mentioned in the second section). In contrast to existing translation systems, the proposed scheme must embody modern linguistic theory so that it provides a more explanatory model of language translation. Furthermore, a restrictive theory of universal grammar, which is necessarily tightly constrained, must be incorporated into the system so that the implementation is facilitated. Finally, the proposed scheme must include several parameters of variation so that it is flexible enough to model translation between several languages.

[12] See Kayne (1981) and Jaeggli (1980).

[13] The details of the *chain* analysis are in Torrego, 1984. They will not be discussed here.

4 Work To Be Done

4.1 Modification of the Marcus Parser

A possible approach to implementing a principle-based translator is to modify the Marcus parser[14] so that it consists of principles whose parameters can be filled with the values that are associated with a particular source or target language. The Marcus model is built to run on a complex set of rules and must be modified to operate on a constrained grammatical theory based on principles and parameters. However, modification of the Marcus parser to embody GB theory in its entirety is an enormous task. As mentioned in Berwick and Weinberg (1984), the Marcus parser can be adjusted to incorporate Subjacency and the relevant predicate of *c-command*. However, whether or not principles such as ECP, Case, Theta, Binding, Bounding, Control, X-Bar and Government can be fully established within the framework of the Marcus parser remains to be seen.

Because the construction of a system which entirely models GB theory is such an enormous task, the Marcus parser modifications will be based on a small range of translation cases (*e.g.*, the Spanish inversion effects discussed by Torrego). Thus, the principles and parameters that will be incorporated into the parser will be based only on those principles and parameters that are required for correct analysis of a certain range of linguistic phenomena.

For example, the Marcus parser builds NP-Aux inversion into the Wh-movement mechanism. However, NP-Aux inversion is an operation that is specific to English. In particular, Spanish does not incorporate NP-Aux inversion into Wh-movement (in fact, as mentioned above, there is no equivalent Aux constituent in Spanish); rather, the operation that takes place in conjunction with Wh-movement is V-Preposing. In order to account for this variation, the operation of Wh-movement would have to be parameterized so that the effects of its operations could be introduced as a parameter value associated with a particular language.

In addition, Spanish inversion effects require that the principles of ECP and Subjacency to be parameterized. As mentioned in the third section, the conditions under which the ECP is violated varies from language to language. First of all, a parameter would be required to account for whether or object extraction is allowed. Secondly, a parameter would be required for encoding whether or not a *chain* is required for proper government. The principle of Subjacency would have to be parameterized with respect to the bounding node associated with a particular language. As shown earlier, certain languages may have S as a bounding node (*e.g.*, English), while others do not (*e.g.*, Spanish and Italian).

Once the Marcus parser is modified so that it incorporates the required principles and parameters, transformation routines will have to be constructed. These routines will take the parsed form (annotated surface structure) returned by the Marcus parser and turn it into a "base form" in which moved constituents have been dropped back into place. This form will be a syntactic structure that can be "inverse-parsed" into the annotated surface form of any target language.

The "inverse Marcus parser" (or generator) will have to be constructed so that it is based on the same principles that the parser uses. Since these principles have been parameterized,

[14] For a description of the Marcus parser, see Marcus, 1980.

the target language parameter values can be dropped into place before the annotated surface structure of the target form is generated.

After the generator has constructed the annotated surface form, the annotations (*i.e.*, the brackets and traces) will be dropped, and the surface structure will be returned.

An example of how the entire process will operate is as follows:

Spanish surface structure:
Qué le dijo Juan a Ricardo?
⇓
Annotated surface structure (after parsing):
$[_{\bar{S}}$ Qué$_i$ $[_S$ le dijo $[_S$ Juan $[_{VP}$ t_v e_i a Ricardo?]]]]
⇓
Base form (after transformation routines):
$[_{\bar{S}}$ $[_S$ Juan $[_{VP}$ le dijo qué a Ricardo]]]
$[_{\bar{S}}$ $[_S$ John $[_{VP}$ said what to Richard]]]
⇓
Annotated surface structure (after generation):
$[_{\bar{S}}$ What$_i$ $[_S$ did John $[_{VP}$ say e_i to Richard]]]
⇓
English surface structure:
What did John say to Richard?

Note that several different processes are taking place at each stage. The parser must know that both Wh-movement as well as V-Preposing have taken place so that it can drop the Wh-trace e_i and the V-trace t_v; the transformation routines must return several moved constituents to their places of origin, and collapse two bracketed S-clauses into one; and the generator must know that Wh-movement took place in the original Spanish sentence so that the corresponding action will take place in the target sentence.

4.2 Difficulties to be Addressed

The toughest part of the construction of the above scheme is deciding how to modify the Marcus parser so that the required principles and parameters are incorporated into the system. The parser will have to be converted from a system that operates on a set of complex, redundant, (somewhat) *ad hoc* rules to one that is based on constrained, modularized subsystems of principles and parameters. The difficulty is that it is not clear how the principles can be incorporated into the system. Furthermore, once the principles have been incorporated, it will be difficult to decide how they will be parameterized. In addition to these complications, it will not be an easy task to determine which principles must be parameterized, and what parametric variations should be allowed. These problems will also arise in the construction of the generator since it is built on the same set of parameterized principles as the parser.

An additional difficulty that must be addressed is that of how to build the generator so that it will "know" what movement rules took place in the original sentence. Since all the traces have been cleaned up during the transformation to "base" form, it appears that some sort of *inverse-trace* will have to be left behind when the transformer moves constituents

back into place. This implies that special purpose routines must be constructed so that these *inverse-traces* can be located, and the corresponding action can be taken.

An alternative approach is to maintain a stack that records the actions that take place during the transformation to "base" form. These operations can then be recovered during the generation process, and the corresponding (inverse) actions can be taken in order to arrive at the annotated surface form. In the translation example of the previous section, the transformation to "base" form would leave one operation in the stack: Wh-replacement (the inverse of Wh-movement). Thus, in the generation stage, the inverse of this operation would be applied. However, the effect of Wh-replacement in Spanish (V-Postposing, which is the inverse of V-Preposing) is not the same as the effect of Wh-replacement in English (NP-replacement, which is the inverse of NP-Aux inversion). Care must be taken so that effects which are not characteristic of a language are forbidden (*e.g.*, V-Preposing should be prohibited in the generation of English). Rather, the parameter associated with the operation must be checked in order to determine what effects are associated with the operation in the target language. Ideally, it should be possible to select certain values of parameters and test whether the resulting structure is well-formed by appealing to the principles of GB.

Despite these difficulties, once the parser and generator designs are chosen, it should be possible to make headway toward reducing the amount of information and time required for machine translation. Ultimately, the system should run on a small and tightly constrained set of parameterized principles.

5 Resources for Research

The research for the Master's degree will be done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support will be provided in the form of a research assistantship under Professor Robert C. Berwick.

The author will make use of an account on the AI Laboratory Computer System: MIT-OZ which operates on the DEC-20. The LISP programming language will be used (MACLISP dialect) and the work may be transferred to lab-owned Lispmachines which use a LISP dialect called Zeta-LISP.

Support for the Laboratory's artificial intelligence research has been provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-80-C-0505.

6 References for Current and Future Research

Berwick, Robert C., and Amy S. Weinberg. (1984). *The Grammatical Basis of Linguistic Performance.* Cambridge, Massachusetts: MIT Press.

Barton, Edward G. Jr. (1984). *Toward a Principle-Based Parser.* Technical Report, A.I. Memo No. 788, MIT, Cambridge, Massachusetts.

Brady, Michael and Robert C. Berwick. (1983). *Computational Models of Discourse.* Cambridge, Massachusetts: MIT Press.

Burzio, Luigi. (1981). *Intransitive Verbs and Italian Auxiliaries.* Doctoral Dissertation, MIT, Cambridge, Massachusetts.

Chomsky, Noam A. (1965). *Aspects of the Theory of Syntax.* Cambridge, Massachusetts: MIT Press.

Chomsky, Noam A and Howard Lasnik. (1977). "Filters and Control," *Linguistic Inquiry* 8:3, 425-504.

Chomsky, Noam A. (1977). "On Wh-Movement,"in Culicover, P.W., and T. Wasow, eds., *Formal Syntax.* New York, New York: Academic Press.

Chomsky, Noam A. (1978). *On Binding.* First Draft of paper, MIT, Cambridge, Massachusetts.

Chomsky, Noam A. (1981). "Principles and Parameters in Syntactic Theory," in N. Hornstein and D Lightfoot, eds., *Explanation in Linguistics.* London England and New York, New York: Longman. 32-75.

Chomsky, Noam A. (1982). *Some Concepts and Consequences of the Theory of Government and Binding.* Cambridge, Massachusetts and London England: MIT Press.

Dorr, Bonnie J. (1983). *Analysis and Critique of MOPTRANS.* Unpublished Paper, Boston University, Boston, Massachusetts.

Jackendoff, Ray S. (1969). *Some Rules of Semantic Interpretation for English.* Doctoral Dissertation, MIT, Cambridge, Massachusetts.

Jackendoff, Ray S. (1972). *Semantic Interpretation in Generative Grammar.* Cambridge, Massachusetts: MIT Press.

Jaeggli, Osvaldo Adolfo. (1980). *On Some Phonologically Null elements in Syntax.* Doctoral Dissertation, MIT, Cambridge, Massachusetts.

Katz, Boris. (1980). *A Three-Step Procedure for Language Generation.* Technical Report, MIT, Cambridge, Massachusetts.

Kayne, Richard S. (1972). "Subject Inversion in French Interrogatives," in J. Casagrande and B. Saciuk, eds., *Generative Studies in Romance Languages.* Rowley, Massachusetts: Newbury House. 70-126.

Kayne, Richard S. and Jean-Yves Pollok. (1978). "Stylistic Inversion, Successive Cyclicity and Move NP in French," *Linguistic Inquiry* 9:4, 595-621.

Kayne, Richard S. (1981). "ECP Extensions," *Linguistic Inquiry* 12, 93-133.

Levin, Beth. (1977). *Mapping Sentences to Case Frames.* Working Paper, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Lytinen, Steven L. and Roger C. Schank. (1982). *Representation and Translation.* Technical Report, Yale University, New Haven, Connecticut.

Lytinen, Steven L. (1983). *Word Disambiguation and Translation.* Technical Report, Yale University, New Haven, Connecticut.

Miller, J. Dale. (1982). *1000 Spanish Idioms.* Provo, Utah: Brigham Young University Press.

Picallo, M. Carme. (1984). "The Infl Node and the Null Subject Parameter," *Linguistic Inquiry* 15:1, 75-102.

Piera, C. (1981). *The Syntax of the Comparative Clauses in Spanish.* m.s., Cornell University, Ithaca, New York.

Plann, Susan. (1982). "Indirect Questions in Spanish," *Linguistic Inquiry* 13:2, 297-312.

Radford, Andrew. (1981). *Transformational Syntax.* Cambridge, London, New York, New Rochelle, Melbourne, Sydney: Cambridge University Press.

Rivero, Maria Luisa. (1978). "Topicalization and Wh Movement in Spanish," *Linguistic Inquiry* 9:3, 513-517.

Salas, Rodrigo. (1971). *Los 1500 Errores Mas Frecuentes de Espanol.* Madrid: Editorial De Vecchi.

Sager, Naomi. (1981). *Natural Language Information Processing: A Computer Grammar of English and Its Applications.* Reading, Massachusetts: Addison-Wesley.

Slocum, Jonathan. (1984). *Machine Translation: its History, Current Status, and Future Prospects.* Working Paper, University of Texas, Austin Texas.

Slocum, Jonathan. (1984). *METAL: The LRC Machine Translation System.* Working Paper, University of Texas, Austin Texas.

Sportiche, Dominique. (1981). "Bounding Nodes in French," *Linguistic Review* 1:2, 219-246.

Stockwell, Robert P., J. Donald Bowen and John W. Martin. (1965). *The Grammatical Structures of English and Spanish.* Chicago Illinois and London England: University of Chicago Press.

Torrego, Esther. (1984). "On Inversion in Spanish and Some of Its Effects," *Linguistic Inquiry* 15:1, 103-129.

Torrego, Esther. (1981). *Spanish as a Pro-Drop Language.* m.s., University of Massachusetts, Boston, Massachusetts.

Wexler, Kenneth and Peter W. Culicover. (1980). *Formal Principles of Language Acquisition.* Cambridge, Massachusetts and London England: MIT Press.

Winograd, Terry. (1983). *Language as a Cognitive Process: Syntax.* Volume 1. Reading, Massachusetts, Menlo Park, California, London, Amsterdam, Don Mills, Ontario, Sydney: Addison-Wesley.

Winston, Patrick Henry and Berthold Klaus Paul Horn. (1984). *LISP.* Second Edition. Reading, Massachusetts, Menlo Park, California, London, Amsterdam, Don Mills, Ontario, Sydney: Addison-Wesley.