A Parameter-Based Message-Passing Parser for MT of Korean and English

DekangLin[§], Bonnie Dorr[†], Jye-hoon Lee[†], Sungki Suh[‡]

[§]Department of Computer Science
University of Manitoba
Winnipeg, MB, Canada, R3T 2N2
lindek@cs.umanitoba.ca

[†]Department of Computer Science
University of Maryland
College Park, MD 20724
{bonnie, jlee}@cs.umd.edu

[‡]Department of Linguistics
University of Maryland
College Park, MD 20742
sksuh@wam.umd.edu

Abstract

The parsing component of previous principle-based machine translation systems are inefficient since they tend to adopt a generate-and-test paradigm. We combine the benefits of a message-passing paradigm with the benefits of a parametric approach in the implementation of a parser that avoids overgeneration and *is* easily ported to multiple languages. The algorithm has been implemented in C++ and successfully tested on well-known, translationally divergent sentences in a MT system called PRINCITRAN.

Keywords: parameter-based MT, message passing, cross-linguistic divergences

1. Introduction

This paper presents an efficient, implemented approach to cross-linguistic parsing for interlingual MT. Our design is based on Government-Binding (GB) Theory (Chomsky, 1981; Haegeman, 1991; van Riemsdijk and Williams, 1986). One of the drawbacks to alternative GB-based parsing approaches is that they generally adopt a filter-based paradigm, generating all possible candidate structures of the sentence that satisfy Xbar theory, and then subsequently applying filters to eliminate those structures that violate GB principles. (See, for example, Abney (1989), Correa (1991), Dorr (1991), Fong (1991), and Frank 1990.) The current approach provides an alternative to filter-based designs which avoids these difficulties by applying principles to *descriptions* of structures without actually building the structures themselves. In effect, structure building is deferred until the descriptions satisfy all principles.

We combine the benefits of the message-passing paradigm with the benefits of the parameterized approach to build a more efficient, but easily extensible system, called PRINCITRAN.[1] Our work extends that of Lin and Goebel (1993) and Lin (1993) in that it provides a parameterization mechanism along the lines of Dorr (1993b) that allows the system to be ported to languages other than English. We focus particularly on the problem of processing head-final languages such as Korean. The algorithm has been implemented in C++ and successfully tested on well-known, translationally divergent sentences.

The next section presents a general framework for parsing by message passing. Section 3 presents our parameterization framework for handling cross-linguistic variation. Section 4 describes our technique for automatic precompilation of parameter settings on a per-language basis. The results of parsing translationally divergent sentences in Korean and English are presented in section 5.

[1] The name PRINCITRAN is derived from the names of two systems, UN1TRAN (Dorr (1993a)) and PRINCIPAR (Lin (1993)).

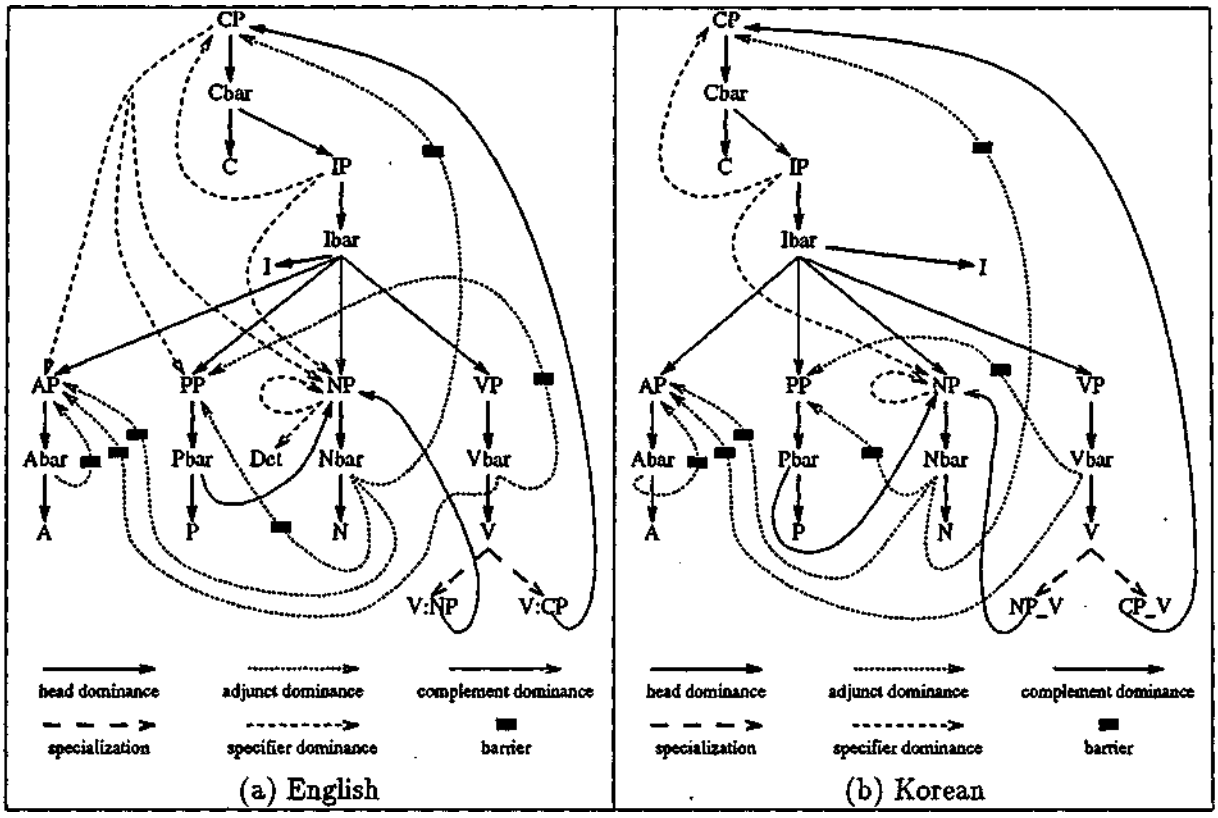

Figure 1: Network Representations of Grammars

We conclude with a discussion about the implications of this approach and its applicability to the problem of multi-language MT.

2. Message Passing Paradigm

Our GB parser is an extension of the message-passing approach proposed by Lin (1993) and Lin and Goebel (1993), which uses a network to encode the grammar. The nodes in the grammar network represent grammatical categories (e.g., NP, Nbar, N) or subcategories, such as V:NP (i.e., a transitive verb that takes an NP as complement). Figure l.a depicts a portion of the grammar network for English.

There are two types of links in the network: subsumption links (e.g., V to V:NP) and dominance links (e.g., Nbar to N). A dominance link from a to β is associated with an integer id that determines the linear order between β and other categories dominated by a, and a binary attribute to specify whether β is optional or obligatory.[2]

Input sentences are parsed by passing messages in the grammar network. The nodes in the network are computing agents that communicate with each other by sending messages in the reverse direction of the links. Each node locally stores a set of items. An item is a triplet that

[2] For the purpose of readability, we have omitted integer id's in the graphical representation of the grammar network; link ordering is indicated instead by the starting points of links, e.g., C precedes IP under Cbar since the link leading to C is to the left of the link leading to IP.

represents a Xbar structure a: **<surface-string, attribute-values, source-messages>**, where **surface-string** is an integer interval [i,j] denoting the i'th to j'th word in the input sentence; **attribute-values** specifies syntactic features of the root node (β); and source-messages is a set of messages that represent immediate constituents of β and from which this item is combined. Each node has a completion predicate that determines whether an item at the node is "complete," in which case the item is sent as a message to other nodes.

When a node receives an item, it attempts to combine the item with items from other nodes to form new items. Two items: $<[i_1, j_1], A_1, S_1>$ and $<[i_2, j_2], A_2, S_2>$ can be combined if

1. their surface strings are adjacent to each other: $i_2 = j_1+1$.
2. their attribute values A_1 and A_2 are unifiable.
3. the source messages come via different links: $links(S_1) \cap links(S_2) = 0$, where links(S) is a function that, given a set of messages, returns the set of links via which the messages arrived.

The result of the combination is a new item: $<[i_1,j_2]>$ unify(A_1, A_2), $S_1 \cup S_2>$. The new item represents a larger Xbar structure resulting from the combination of the two smaller ones. If the new item satisfies local constraints, it is considered valid and saved in the local memory. Otherwise, it is discarded. A valid item satisfying the completion predicate of the node is sent further as a message to other nodes. Details of the algorithm are provided in Lin (1993).

3. Parameterization for Cross-Linguistic Variation

The structure of the grammar network in section 2 is too language-specific to be applicable to languages other than English. The most obvious flaw with this network is that each dominance link is associated with an integer id that determines the linear order of phrasal constituents. In this particular network, all phrasal heads precede their complements. However, in a head-final language such as Korean, the reverse order is required. In order to capture this distinction, we incorporate the parameterization approach of Dorr (1993b) into the message-passing framework so that the grammar network can be automatically precompiled on a per-language basis.

The reason the message-passing paradigm is so well-suited to a parameterized model of language parsing is that, unlike head-driven models of parsing, the main message-passing operation is capable of combining two nodes (in any order) in the grammar network. The result is that a head-final language such as Korean is as efficiently parsed as a head-initial language such as English. What is most interesting about this approach is that model is consistent with experimental results (see, for example, Suh (1993)) which suggest that constituent structure is computed prior to the appearance of the head in Korean.

The remainder of this section describes our approach to parameterization of certain subtheories of Government-Binding (GB) Theory; we conclude with a summary of the syntactic parameter settings for English and Korean.

3.1. Xbar Theory

Xbar theory assumes that a constituent order parameter is used for specifying phrasal ordering on a per-language basis:

(1) **Constituent Order:** The relative order between the head and its complement can vary, depending on whether the language in question is (i) head-initial or (ii) head-final.

The structure above represents the relative order observed in Korean, i.e., the head-final parameter setting (ii). In English, the setting of this parameter is (i). This ordering information is encoded in the grammar network by virtue of the relative ordering of integer id's associated with network links. Other types of parameters encoded in the grammar network are those pertaining to basic categories, pre-terminal categories (e.g., determiner), potential specifiers, and adjuncts for each basic category.

3.2. Trace Theory

In general, NP and CP nodes are considered to be barriers to movement. However, Korean allows the head noun of a relative clause to be construed with the empty category across more than one intervening CP node, as shown in the following:

(2) [CP [CP t_1 t_2 kyengyengha-ten] hoysa$_2$-ka manghayperi-n] Bill$_1$-un yocum uykisochimhay issta
managed-Rel company-Norn is bankrupt-Rel -Top these days is depressed
'Bill is such a person that the company which was managed by him has been bankrupt, and he is depressed these days'

The subject NP 'Bill' is coindexed with the trace in the more deeply embedded relative clause. If we assume, following Chomsky (1986a), that relative clause formation involves movement from an inner clause into an outer subject position, then the grammaticality of the above example suggests that the Trace theory must be parameterized so that crossing more than one barrier is allowed in Korean. Our formulation of this parametric distinction is as follows:

(3) **Barriers:** (i) only one crossing permitted; (ii) more than one crossing permitted.

In English the setting would be (i); in Korean the setting would be (ii).

3.3. Case Theory

In general, it is assumed that the relation between a case assigner and a case assignee is biunique. However, this assumption rules out so-called multiple subject constructions which are commonly used in Korean:

(4) John-i phal-i pwureciessta
-Nom arm-Nom was broken
'John is in the situation that his arm has been broken'

The grammaticality of the above example suggests that Nominative Case in Korean must be assigned by something other than tensed Infl. Thus, we parameterize Case Assignment as follows:

(5) **Case Assignment**: Accusative case is assigned by transitive V; Nominative case is assigned by (i) tensed Infl; (ii) IP predication.

In a biunique case-assignment language such as English, the setting for Nominative case assignment would be (i); in Korean, the setting would be (i) and (ii).

3.4. Summary of Parameter Settings for English and Korean

We have just seen that certain types of syntactic parameterization may be captured in the grammar network (e.g., Xbar parameters such as constituent order). In addition to these, there are syntactic parameters that must be programmed into the message-passing mechanism itself, not just into the grammar network.

Sub-Theory	Parameter	English Setting	Korean Setting
X	Basic Categories	C I V N P A	C I V N P A
	Pre-terminals	ADV NUM DET	ADV DEM
	Constituent Order	I: SPEC-INITIAL HEAD-INITIAL N: SPEC-INITIAL HEAD-INITIAL C: SPEC-INITIAL HEAD-INITIAL A: HEAD-INITIAL P: HEAD-INITIAL V: HEAD-INITIAL	I: SPEC-INITIAL HEAD-FINAL N: SPEC-INITIAL HEAD-FINAL C: HEAD-FINAL A: HEAD-FINAL P: HEAD-FINAL V: HEAD-FINAL
	Specifiers	I: NP C: NP, PP, AP N: NP (+gen), DET	I: NP N: NP (+gen), DEM
	Adjunction	Ibar: PP (left), ADV (right), NP (right) Vbar: PP (right), ADV (left) Nbar: AP (left), PP (right), CP (right), NUM (left) Abar: ADV (left)	Ibar: PP (left), ADV (left), NP (left) Vbar: PP (left), ADV (left) Nbar: AP (left), PP(+gen) (left), CP (left) Abar: ADV (left)
Trace	Barriers	only one crossing permitted	more than one crossing permitted
Case	Case Assignment	Nominative: tensed Infl; Accusative: transitive V	Nominative: tensed Infl, IP predication; Accusative: transitive V

Figure 2: Syntactic Parameter Settings for Korean

Figure 2 shows the syntactic parameter settings for English and Korean. The English settings are drawn from Dorr (1993b). The same paradigm was followed in our analysis of Korean parameters. The remainder of this paper will focus on how we automatically precompile the English and Korean parameter settings concerning Xbar theory into the grammar network (i.e., Basic Categories, Pre-tenninals, Constituent Order, Specifiers, and Adjunction).

4. Parameter Compilation Algorithm

Our algorithm for automatic precompilation of the parameter settings into a grammar network consists of dynamic generation of code that is read as data by the message-passing parsing system. The following steps are used to construct the code:

1. Use **`defcategory`** to define:
   - X0 nodes (using Basic Categories and Constituent Order parameters)
   - Barl and Bar2 nodes (projecting from X0 nodes)
   - Pre-terminal nodes (using Pre-terminals parameter)
2. Use **`defeature`** to define:

- Specifier links (using Specifiers parameter)
- Adjunct links (using Adjunction parameter)

Figure l.b shows the network that is generated as a result of executing this algorithm using the Korean Xbar parameter settings. A portion of the code that is automatically generated for this network is given in Appendix A.

5. Results of Running Test Sentences for Korean/English

The parameterized framework described above has been implemented in C++ and successfully tested on well known, translationally divergent sentences. Dorr (1990) describes the problem of MT divergences in English, Spanish, and German. We present analogous examples here for English and Korean:

Structural Divergence:
E: John married Sally
K: John-i Sally-wa kyelhonhayssta
-Nom -with married
'John married with Sally'

Conflational Divergence:
E: John helped Bill
K: John-i Bill-eykey towum-ul cwuessta
-Nom -Dative help-Acc gave
'John gave help to Bill'

Categorial Divergence:
E: John is fond of music
K: John-un umak-ul coahanta
-Top music-Ace like
'As for John, (he) likes music'

We ran the parameterized parser on both the English and Korean sentences shown here. The results shown in Figure 5 which were obtained from running the program on a Sparcstation ELC.[3] In general, the times demonstrate a speedup of 2 to 3 orders of magnitude over previous principle-based parsers on analogous examples such as those given in Dorr (1993a). Even more significant is the negligible difference in processing time between the two languages, despite radical differences in structure, particularly with respect to head-complement positioning. This is an improvement over previous parameterized approaches in which cross-linguistic divergences frequently induced timing discrepancies of 1-2 orders of magnitude due to the head-initial bias that underlies most parsing designs.

	Parse	Time
E:	[CP [Cbar [IP [NP [Nbar [N John]]] [Ibar [VP [Vbar [V_NP married] [NP [Nbar [N Sally]]]]]]]]	.15 sec.
K:	[CP [Cbar [IP [NP [Nbar [N John-i]]] [Ibar [VP [Vbar [PP [Pbar [NP [Nbar [N Sally]]] [P wa]]] [V_PP kyelhonhayssta]]]]]]	.12 sec.
E:	[CP [Cbar [IP [NP [Nbar [N John]]] [Ibar [VP [Vbar [V_NP helped] [NP [Nbar [N Bill]]]]]]]]	.10 sec.
K:	[CP [Cbar [IP [NP [Nbar [N John-i]]] [Ibar [VP [Vbar [PP [Pbar [NP [Nbar [N Bill]]] [P eykey]]] [NP [Nbar [N towum-ul]]] [V_PP_NP cwuessta]]]]]]	.19 sec.
E:	[CP [Cbar [IP [NP [Nbar [N John-un]]] [Ibar [VP [Vbar [V_AP is] [AP [Abar [A fond] [PP [Pbar [P of] [NP [Nbar [N music]]]]]]]]]]]	.12 sec.
K:	[CP [NP [0] [Nbar [N John-un]]] [Cbar [IP t [0] [Ibar [VP [Vbar [NP [Nbar [N umak-ul]]] [V_NP coahanta]]]]]]	.07 sec.

Figure 3: Parameterized Parsing of English and Korean Divergence Examples

[3] Certain intermediate network nodes (e.g., Nspec and Nadj) do not show up in the output of the parser.

6. Implications

We are currently incorporating the parameterized parser into an interlingual MT system called PRINCITRAN. The current framework is well-suited to an interlingual design since the linking rules between the syntactic representations given above and the underlying lexical-semantic representation are well-defined (see Dorr (1993a)). We adopt the Lexical Conceptual Structure (LCS) of Dorr's work and use a parameter-setting approach to account for the divergences presented in the last section. (Dorr (1993b) describes a parametric approach to mapping between the interlingua and the syntactic structure.)

A preliminary investigation has indicated that the message-passing paradigm is useful for generation as well as parsing, thus providing a suitable framework for bidirectional translation. The algorithm for generation is similar to that of parsing in that both construct a syntactic parse tree over an unstructured or partially structured set of lexical items. The difference is characterized as follows: in parsing, the inputs are sequences of words and the output is a structure produced by combining two adjacent trees into a single tree at each processing step;[4] in generation, the inputs are a set of unordered words with dependency relationships derived from the interlingua (LCS). The generation algorithm must produce structures that satisfy the same set of principles and constraints as the parsing algorithm.

Three areas of future work are relevant to the current framework: (1) scaling up the Korean dictionary, which currently has only a handful of entries for testing purposes;[5] (2) the installation of a Kimmo-based processor for handling Korean morphology;[6] and (3) the incorporation of non-structural parameterization (i.e., parameters not pertaining to X theory such as Barriers and Case Assignment).

Summarizing, we have shown that the parametric message-passing design is an efficient and portable approach to parsing. We have automated the process of grammar-network construction, and have demonstrated that the system handles well-known, translationally divergent sentences. We expect that the current framework is suitable for bidirectional, interlingual MT since the message-passing paradigm may be used for generation as well as parsing.

Acknowledgements

Dekang Lin was supported by Natural Sciences and Engineering Research Council of Canada grant OGP121338. Bonnie Dorr and her students, Jye-hoon Lee and Sungki Suh, have been partially supported by the Army Research Office under contract DAAL03-91-C-0034 through Batelle Corporation, by the National Science Foundation under grant IRI-9120788 and NYI IRI-9357731, and by the Army Research Institute under contract MDA-903-92-R-0035 through Microelectronics and Design, Inc.

---

[4] The LCS composition routine described in Dorr (1992) derives the interlingua from the resulting syntactic representation.

[5] Our English dictionary has 90K entries, constructed automatically by applying a set of conversion routines to OALD entries. We have begun negotiations with the LDC for the acquisition of a Korean MRD for which we intend to construct similar routines.

[6] The English dictionary used by the message-passing system contains all morphological derivatives of every word. This approach would be impractical for Korean since the morphology is significantly richer.

References

1. Abney, S. (1989). "A Computational Model of Human Parsing," Journal of Psycholinguistic Research, 18, 129-144.
2. Billot, *S.* and B. Lang (1989). The structure of shared forests in ambiguous parsing. In *Proceedings of ACL-89,* 143-151.
3. Chomsky, N. (1981). *Lectures on Government and Binding.* Foris Publications, Cinnaminson, USA.
4. Correa, N. (1991). Empty categories, chains, and parsing. In Berwick, R. C., Abney, S. P., and Tenny, C., editors, *Principle-Based Parsing: Computation and Psycholinguistics,* 83-121. Kluwer Academic Publishers.
5. Dorr, B. (1990). Solving thematic divergences in machine translation. In *Proceedings of ACL-90,* 127-134. University of Pittsburgh, Pittsburgh, PA.
6. Dorr, B. (1991). Principle-based parsing for machine translation. In Berwick, R. C., Abney, S. P., and Tenny, C., editors, *Principle-Based Parsing: Computation and Psycholinguistics,* 153-184. Kluwer Academic Publishers.
7. Dorr, B. (1992). "The use of lexical semantics in interlingual machine translation," Machine Translation, 7:3, 135-193.
8. Dorr, B. (1993a). *Machine Translation: A View from the Lexicon,* MIT Press, Cambridge, MA.
9. Dorr, B. (1993b). "Interlingual machine translation: a parameterized approach," Artificial Intelligence, 63:1&2, 429-492.
10. Fong, S. (1991). The computational implementation of principle-based parsers. In Berwick, B. C., Abney, S. P., and Tenny, C., editors, *Principle-Based Parsing: Computation and Psycholinguistics,* 65-82. Kluwer Academic Publishers.
11. Frank, R, (1990). Licensing and Tree Adjoining Grammar in GB Parsing. In *Proceedings of ACL-90,* 111-118. University of Pittsburgh, Pittsburgh, PA.
12. Haegeman, L. (1991). *Introduction to Government and Binding Theory.* Basil Blackwell Ltd.
13. Lin, D. and R. Goebel (1993). Context-free grammar parsing by message passing. In *Proceedings of PACLING-93,* Vancouver, BC.
14. Lin, D. (1993). Principle-based parsing without overgeneration. In *Proceedings of ACL-93,* 112-120. Columbus, Ohio.
15. Suh, S. (1993). How to process constituent structure in head final languages: The case of Korean. In *Proceedings of Chicago Linguistic Society,* 29.
16. Tomita, M. (1986). *Efficient Parsing for Natural Language.* Kluwer Academic Publishers, Norwell, Massachusetts.
17. van Riemsdijk, H. and E. Williams (1986). *Introduction to the Theory of Grammar.* Current Studies in Linguistics. The MIT Press, Cambridge, Massachusetts.

A. Automatically Precompiled Code for Korean Grammar Network

```
(defcategory I Bar0Node
   (default-atts ((cat i) -head-initial)))
(defcategory N Bar0Node
   (default-atts ((cat n) -head-initial)))
(defcategory V Bar0Node
   (default-atts ((cat v) -head-initial)))
(defcategory P Bar0Node
   (default-atts ((cat p) -head-initial)))
(defcategory Ibar Bar1Node (default-atts ((cat i))))
(defcategory Nbar Bar1Node (default-atts ((cat n))))
(defcategory Vbar Bar1Node (default-atts ((cat v))))
(defcategory Pbar Bar1Node (default-atts ((cat p))))
```

```
(defcategory IP Bar2Node
   (default-atts ((cat i) +spec-initial)))
(defcategory NP Bar2Node
   (default-atts ((cat n) +spec-initial)))
(defcategory VP Bar2Node (default-atts ((cat v))))
(defcategory PP Bar2Node (default-atts ((cat p))))
(defcategory DEM DEMNode (default-atts ((cat DEM))))
(defeature IP (NP spec))
(defeature NP (NP spec (att-filter +genitive)) (DEM spec))
(defeature Nbar Nadj
      (PP adjunct left (att-filter +genitive) (+optional))
      (AP adjunct left (+optional))
      (CP adjunct left (+optional)))
(defeature Abar Aadj (ADV adjunct left (+optional)))
```