7th Summer School of the German Linguistic Society/
7. Sommerschule der DGfS

Walter Bisang, Mainz
wbisang@mail.uni-mainz.de

Typology 8

Motivations I:
Parsing (Hawkins 1994)

1. The basic idea of Hawkins (1994)

1.1. The Constituent Recognition Domain

Words and constituents are positioned relative to each other in a way that allows optimal recognition of a given syntactic structure and its immediate constituents, that is, word-order patterns in the world's languages are structured in such a way that the human parser can recognize the whole structure of a given syntactic structure with its immediate constituents as soon as possible.

This principle is at work in grammar as well as in performance (the actual use of a grammar). Since we shall deal with universals of word order, I shall concentrate on the way in which optimal parsing conditions are reflected in grammar and on the word-order types that are attested in the grammars of the world's languages. The following example, however, belongs to performance. It is known under the term of *Heavy NP Shift*:

(1) a. I VP[gave NP[the valuable book that was extremely difficult to find] PP[to Mary]]

1 2 3 4 5 6 7 8 9 10 11

b. I VP[gave PP[to Mary] NP[the valuable book that was extremely difficult to find]]

1 2 3 4

The VP in the above sentence (1) consists of three immediate constituents:

V *gave*
NP *the valuable book that was extremely difficult to find*
PP *to Mary*

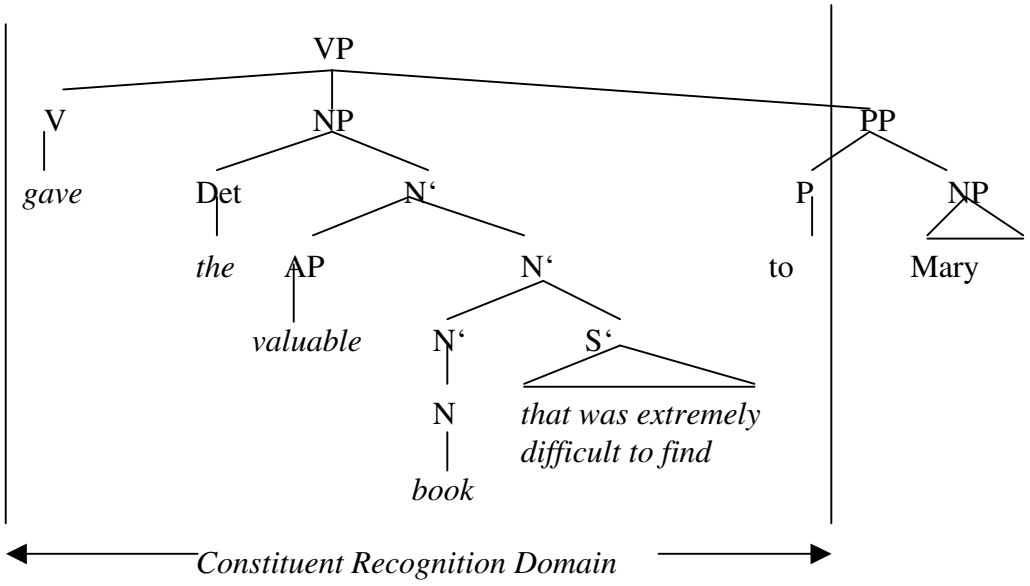
Sentence (1a) follows the normal word-order rules of English, that is, V NP PP. Given the lengths of the NP, the human parser has to wait a long time until it finally arrives at the eleventh word, that is, the preposition of the PP which allows it to see the entire structure of the VP. If the heavy (= long) NP is moved to the end of the VP the whole structure of the VP can be recognized much earlier. As soon as the parser arrives at the fourth word (*the*) it can recognize the entire structure of the VP with its three immediate constituents (V, NP, PP). Heavy NP Shift is thus a strategy to improve parsing conditions.

The basis for calculating optimal parsing conditions is the *Constituent Recognition Domain* which is defined as follows:

(2) *Constituent Recognition Domain* (CRD)
The CRD for a phrasal mother node M consists of the set of terminal and non-terminal nodes that must be parsed in order to recognize M and all ICs of M, proceeding from the terminal node in the parse string that constructs the first IC on the left, to the terminal node that constructs the last IC on the right, and including all intervening terminal nodes and the non-terminal nodes that they construct. (Hawkins 1994: 58 - 59)

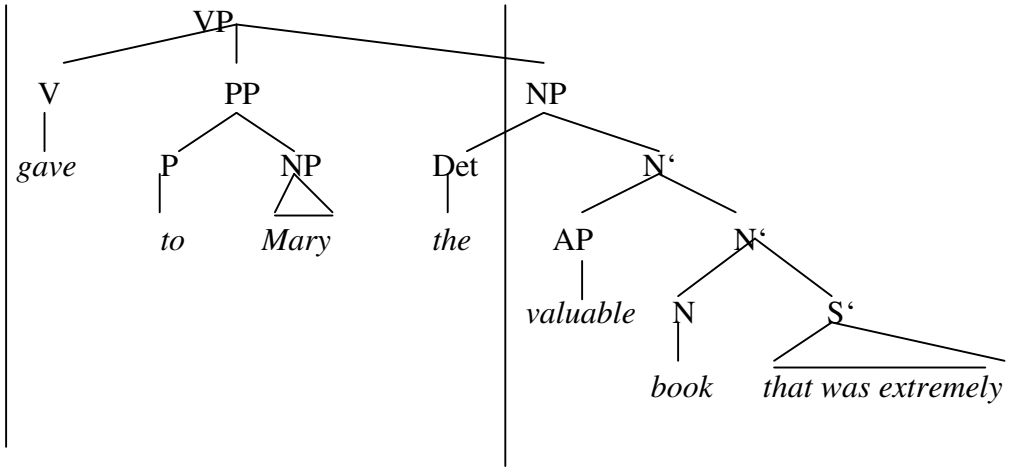
According to this definition, the *Constituent Recognition Domain* of the VP in example (1a) is <*gave the valuable book that was extremely difficult to find to*>, whereby the verb give indicates that we are dealing with a VP. The IC structure of this VP becomes clear with the appearance of the P *to*.

(1) a'.



The Constituent Recognition Domain of (1b) is <*gave to Mary the*>:

(1) b.



difficult to find

←*Constituent Recognition Domain*→

The smaller/the less complex the Constituent Recognition Domain is the better for parsing, that is, the sooner will the parser be able to recognize the whole immediate-constituent structure of a superordinate construction. From this perspective, the Constituent Recognition Domain is the basis on which preferred word orders (in the grammar of individual languages as well as in performance) can be calculated.

1.2. Parsing in head-initial and head-final languages

The above example is from English, a VO language. What happens with OV languages/verb-final languages? If we take the VP as an example, the head, that is V, turns up at the very end of the VP. The parser first sees the NP and cannot be sure about the syntactic status of that NP until it gets to the verb. Of course, this problem arises in each case where we have phrase-final heads. In Hawkins' view, the parser puts constituents whose syntactic status is not clear in a **look-ahead buffer**, that is, a kind of „waiting room" where the constituent remains until the parser is able to detect its syntactic status. The following example from Japanese is to illustrate this:

(3) Japanese (Hawkins 1994: 66)

a. S_1[$_{NP}$[Mary ga] $_{VP}$[$_{S'}$[$_{S2}$[kinoo Johnga kekkon shi-ta] to] it-ta]]
Mary NOM yesterday JohnNOM marry-PST QUOT say-PST

b. S_1[$_{S'}$[$_{S2}$[Kinoo John ga kekkon shi-ta] to] $_{NP}$[Mary ga] $_{VP}$[it-ta]]
yesterday John NOM marry-PST QUOT Mary NOM say-PST

'Mary said thad John married yesterday.'

In both of the above examples, a clause S_1 *Mary ga itta* ‚Mary said' and a second clause S_2 *kinoo John ga kekkon sita* ‚yesterday John married' are linked by the quotational particle *to* in such a way that S_2 is an immediate constituent of S_1. In (3a), the subject of S_1 *Mary ga* is separated from the rest of S_1 by S_2 as a whole. In (3b) S_1 is not interrupted by S_2. We have first S_2 which is marked by *to* at its end followed by S_1.

In (3a), the Constituent Recognition Domain for the sentence S_1 is very large, because the parser has to go through the whole of S_1 beginning with *Mary* up to *itta* ‚said' to recognize the IC structure of that sentence. If S_2 is preposed to S_1 followed by the quotational particle *to*, the Constituent Recognition Domain gets considerably shorter. As soon as the parser arrives at *to*, it recognizes the preceding clause S_2 as a constituent of S_1, the next NP *Mary ga* will be stored in the look-ahead buffer until *itta*'said' determines the syntactic status of that NP and of the whole sentence. Thus, <*to Mary ga itta*> is the Constituent Recognition Domain of (3b). Notice that the second clause S_2 is constructed as an immediate constituent of S_1 only when the parser arrives at the quotational particle *to*. Before that time there is no final decision concerning the syntactic status of S_2 — it will be kept in the look-ahead buffer until its status becomes clear. In fact in Japanese, the parser very often can come up with a final decision about the status of individual constituents only at the very end of a sentence.

In a head-initial language (a language in which the head of a phrase is at its beginning) such as English, the parser starts at the beginning of an utterance where the relevant information for the overall structure of a constituent is provided, in head-final languages such as Japanese the parser first has to go through a number of lower constituents which will be stored in the look-ahead buffer until finally the information concerning the immediate constituent-structure of the higher phrases becomes visible.

1.3. Calculating optimal word-order constellations: Early Immediate Constituents

The Principle of Early Immediate Constituents formulates the correlation between the Constituent Recognition Domain and preferred word orders. The basic idea is that the human parser prefers word orders whose ratio between immediate constituents within the Constituent Recognition Domain and the number of non-immediate constituents (IC-to-non-IC ratio) is relatively high. In other words, the higher the IC-to-non-IC ratio of a syntactic structure is the higher is the degree of its preference.

(4) *Early Immediate Constituents (EIC)*
The human parser prefers linear orders that maximize the IC-to-non-IC ratios of constituent recognition domains. (Hawkins 1994: 77)

In the above example (1a) there are 3 immediate constituents and 28 terminal and non-terminal nodes which are not immediate constituents of the VP in Hawkins‘ (1994) calculation (my structural analysis in [1a‘] deviates from that of Hawkins‘). Thus, we get an IC-to-non-IC ratio of 3/28 = 0.107 or 10.7% for (1a). In the case of (1b), there are again 3 immediate constituents but only 8 terminal and non-terminal nodes which are not immediate constituents of the VP. Thus, we get an IC-to-non-IC ratio of 3/8 = 0.375 or 37.5%. Since the IC-to-non-IC ratio of (1b) is higher than that of (1a), (1a) is preferred. Thus, Heavy NP Shift is accepted in the case of (1b) because this yields a higher IC-to-non-IC ratio.

For the sake of completeness, this is the way in which Hawkins‘ (1994) calculates IC-to-non-IC ratios in general:

(5) *Calculating IC-to-non-IC ratios*
The IC-to-non-IC ratio for a CRD is calculated by dividing the number of ICs in the domain by the total number of non-ICs (or words alone) in that domain, expressing the result as a percentage. The ratio for a whole sentence is the aggregate of the scores for all CRDs within the sentence. (Hawkins 1994: 76 - 77)

The EIC Principle is **gradual**. It predicts optimal word-order sequences and it also predicts the degree of preference of non-optimal word-order sequences depending on their IC-to-non-IC ratios.

(6) *EIC* (Expanded)
The human Parser prefers linear orders that maximize the IC-to-non-IC ratios of constituent recognition domains. Orders with the most optimal ratios will be preferred over their non-optimal counterparts in the unmarked case; orders with non-optimal ratios will be more or equally preferred in direct proportion to the magnitude of their ratios. For finer discriminations, IC-to-non-IC ratios can be measured left-to-right.

(Hawkins 1994: 78 - 79)

1.4. The EIC Principle and its typological relevance

On the basis of the gradual character of the Principle of Early Immediate Constituents it is possible to establish typological hierarchies of the following type: If a word-order sequence with a lower IC-to-non-IC ratio ispossible, sequences with a higher ratio belonging to the same domain will also be accepted. §2 on the Prepositional Noun Modifier Hierarchy is to illustrate this.

2. The EIC Principle and the Prepositional noun Modifier Hierarchy

The basic facts again:

(7) Prepositional Noun Modifier Hierarchy (PrNMH):
If a language is prepositional, then if RelN then GenN, if GenN then AdjN, and if AdjN then DemN. (Hawkins 1994: 316)

(8) Rel < Gen < Adj < {Dem, Num}

(9) Hawkins (1994: 316):

a.	Prep:	ø	[NDem, NAdj, NGen,	NRel]	z.B.: Arabic, Thai
b.	Prep:	DemN	[NAdj,NGen,	NRel]	z.B.: Masai, Spanish
c.	Prep:	DemN, AdjN	[NGen,	NRel]	z.B.: Greek, Maya
d.	Prep:	DemN, AdjN, GenN		[NRel]	z.B.: Maung
e.	Prep:	DemN, AdjN, GenN, RelN		ø	z.B.: Amharic

If we look at the syntactic structure of the Modifier-Noun constructions in (9), we get the following generalization:

(10) $_{PP}$[P $_{NP}$[___ N ...]

The lower-case line in (10) represents a modifying category which can occur in this position, that is, Dem, Adj, Gen or Rel.

(10‘) $_{PP}$[P $_{NP}$[___ N ...]

a. ø
b. Dem
c. Adj
d. PossP (= Gen)
e. S' (= Rel)

Since the modifier is inserted within the P and the NP of the PP as a whole, all word-order constellations in (10‘) are less optimal than those in which the modifier follows the noun. Constellations of the type in (10‘) are called **center embedding**, because a lower constituent is inserted in the center of a higher constituent in such a way that parts of the higher constituent occur to the left as well as to the right of that constituent. Central embedding generally leads to

the least optimal structures. Hawkins makes the following predictions concerning immediate constituents which are center-embedded into a Constituent Recognition Domain:

(11) *EIC Grammatical Hierarchy Prediction*

Given: an IC position P center-embedded in a CRD for a phrasal node D (i.e. there is non-null material to both left and right of P within D);

a set of alternative categories {C} that could occur in position P according to the independently motivated and unordered PS-rules of the grammar.

Then: if a category C_i from {C} produces low EIC ratios for D and is grammatical, then all the other categories from {C} that produce improved ratios will also be grammatical. (Hawkins 1994: 102, 315 - 316)

This prediction can be tested with the Prepositional Noun Modifier Hierarchy. As we can see from table 1 below, the IC-to-word ratio[1] of the structures with center embedding decreases from DemN to AdjN to PossN to S'N. On the other hand, all the word-order types in which the modifier follows the noun (NDem, NAdj, NPoss, NS') have optimal IC-to-word ratios of 1. (The categories which are center-embedded are in bold print in table 1. Hawkins' (1994) calculation is based on the assumption that Dem and Adj consist of 1 word, PossP of 2 words and S' of 4 words.)[2]

Structure	IC-to-word ratio within CRD of PP	Number of languages with this word order (total: 61 languages)	Percentage of languages with this word order (100% = 61 languages)
1. PP[P NP[**Dem** N]] |_____| 2/2	100%	29	48%
2. PP[P NP[**Adj** N]] |_____| 2/3	67%	17	28%
3. PP[P NP[**PossP** N]] |_____| 2/4	50%	8	13%
4. PP[P NP[**S'** N]] |_____| 2/6	33%	1	2%
1'. PP[P NP[N Dem]] |____|	100%	32	52%

[1] Since it is often difficult to see the phrase structure of a given constituent from the grammars written on individual languages, Hawkins often simplifies the IC-to-non-IC ratio by comparing the number of immediate constituents to the number of words within the Constituent Recognition Domain. In this case, Hawkins uses the term **IC-to-word ratio**.

[2] The calculation of the IC-to-word ratio in table 1 is calculated as follows:

(i) $$\text{IC-t-word ratio} = \frac{\text{number of IC within CRD}}{\text{number of words within CRD}}$$

2/2			
2'. PP[P NP[N Adj]]	100%	44	72%
|_____|			
2/2			
3'. PP[P NP[N PossP]]	100%	53	87%
|_____|			
2/2			
4'. PP[P NP[N S']]	100%	60	98%
|_____|			
2/2			

The above table 1 confirms Hawkins prediction in (11). The number of languages with central embedding of a modifier within the PP decreases parallel to the decrease of the IC-to-word ratio of the respective structure.

3. Basic word order and parsing (a survey)

VO (SVO, VSO, VOS) with the existence of a VP:[3]

Word-order type	average IC-t-word ratio	number of languages in Tomlin (1986)	percentage within VO languages
S[mS VP[V mO]][4]			
CRD of S: 2/3 = 67%	84%	168	77%
CRD of VP: 2/2 = 100%			
S[VP[V] mS VP[mO]]			
CRD of S: 2/2 = 100%	75%	37	17%
CRD of VP: 2/4 = 50%			
S[VP[V mO] mS]			
CRD of S: 2/5 = 40%	70%	12	6%
CRD of VP: 2/2 = 100%			

Table 2 from Hawkins (1994: 331)

OV (SOV, OVS, OSV) with the existence of a VP:

Word-order type	average IC-t-word ratio	number of languages in Tomlin (1986)	percentage within VO languages
S[Sm VP[Om V]]			
CRD of S: 2/3 = 67%	84%	180	97%

[3] The calculations below are based on the assumption that in VO languages the V constituent consists of 1 word, the S constituent of 2 words and the O constituent of 3 words. For OV languages, the S constituent consists of 3 words, the O constituent of 2 words and the V constituent of 1 word.

[4] The letter „m" indexed to S and O marks where the head of the constituent is situated. E.g. mO means object constituent whose head is on the left, O_m means object constituent whose head is on the right.

CRD of VP: 2/2 = 100%				
$S[_{VP}[O_m\ V]\ S_m]$				
CRD of S: 2/5 = 40%	70%		5	3%
CRD of VP: 2/2 = 100%				
$S[_{VP}[O_m]\ S_m\ _{VP}[V]]$				
CRD of S: 2/4 = 50%	45%		0	0%
CRD of VP: 2/5 = 40%				

Table 3 from Hawkins (1994: 335)

For languages with a flatter sentence structure, that is, for languages with no VP, the situation is more simple:

VO languages	OV languages
$S[_mS\ V\ _mO]$: 75%	$S[S_m\ O_m\ V]$: 75%
$S[V\ _mS\ _mO]$: 75%	$S[O_m\ V\ S_m]$: 60%
$S[V\ _mO\ _mS]$: 60%	$S[O_m\ S_m\ V]$: 60%

Summary of the above tables:

	with VP	without VP
$_mS\ V\ _mO$	**84%**	75%
$V\ _mS\ _mO$	**75%**	**75%**
$V\ _mO\ _mS$	**70%**	60%
$S_m\ O_m\ V$	**84%**	75%
$O_m\ V\ S_m$	**70%**	60%
$O_m\ S_m\ V$	45%	**60%**

Table 4 from Hawkins (1994: 338)

We can take whatever IC-to-word ratio from the above tables, in each case it parallels the percentage of frequency with which that particular type is represented in the world's languages as calculated by Tomlin (1986):

IC-to-word ratio:	84%	75%	70%	60%
	SVO SOV	> VSO	> VOS OVS	> OSV
Percentage of the world's lxs in Tomlin (1986):	87%	9%	4%	0%

References

Hawkins, John A. 1994. *A performance theory of order and constituency*. Cambridge: Cambridge University Press.

Hawkins, John A. 1998. „Some isssues in a performance theory of word order“, in: Siewierska, Anna. Ed. *Constituent order in the languages of Europe*, 729 – 781. Berlin: Mouton de Gruyter.

Hawkins, John A. „Processing complexity and filler-gap dependencies across grammars“, in: *Language* 75, 244 – 285.