Simplest Merge

Jan-Wouter Zwart
University of Groningen

Seminari de lingüística a la Universitat Autónoma de Barcelona
January 16, 2007

1. Merge

(1) "...the most elementary property of language (...) is that it is a system of discrete infinity consisting of hierarchically organized objects. Any such system is based on an operation that takes *n* syntactic objects already formed and constructs from them a new [syntactic object]. Call the operation *Merge*." (Chomsky 2005: 4)

(2) Needed for any system generating discrete infinity (Merge)
a. a set of elements (=Numeration N, Resource)
b. an operation on members of the Numeration

(3) What is in the Numeration N?
Ideally, no restrictions, i.e. any type of linguistic object (feature, morpheme, word, phrase) (cf. Ackema & Neeleman 2005)

a. *morpheme* + *word*

werk-er	'worker'
work-AG	

b. *morpheme* + *phrase*

[kat	uit	de	boom	kijk]-	er-	ij
cat	out	the	tree	look	AG	ABSTR

'the activity of being cautious, wait-and-see attitude'

c. *morpheme* + *clause*

[ban	de	bom]-	er
ban	the	bomb	AG

'person involved in anti-bomb activities'

d. *noun* + *clause*

[doe	dat	nou	niet]-	houding
do	that	PRT	not	attitude

'attitute of advising caution'

(4) What is the operation?
Chomsky 1995, Collins 1998: 1. select
2. merge (=combine in a set)

(5) "arguably restriction of computational resources limits *n* for Merge to two"

"A natural requirement for efficient computation is a 'no tampering condition' NTC: Merge of X and Y leaves the two [syntactic objects] unchanged. If so, then Merge of X and Y can be taken to yield the set {X, Y}, the simplest possibility worth considering."

"A more complex alternative, consistent with NTC, is that Merge forms the pair 〈X, Y〉. The underlying issue is whether linear order plays a role in narrow syntax and mapping to [the semantics interface] C-I, or whether it is restricted to the phonological component."

(Chomsky 2005: 5)

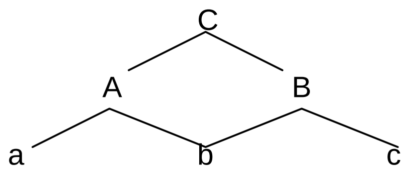
(6) Confusion a. order (mathematics, = asymmetric) *as in* ordered pair
b. order (phonology, = precedence in time) *as in* word order

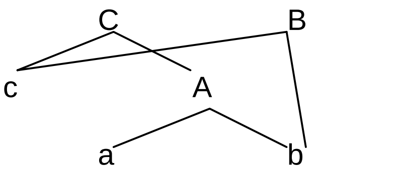
(7) Question: is order (6b) a function of order (6a), or independent of it?
(are interpretive effects at C-I likewise a function of order [6a] or not?)

(8) Implicit in most implementations of Merge:

one of elements selected is the 'current derivation' = D

if not:	*interarboreal operations*			or	*countercyclic operations*		
	merge a, b	->	{a,b} = A		merge a, b	->	{a,b} = A
	merge b, c	->	{b,c} = B		merge c, A	->	{c,A} = C
	merge A,B	->	{A,B} = C		merge b, c	->	{b,c} = B





(9) inherent asymmetry (**except first merge**)
1a. select $\alpha \in$ N
1b. select D (not: select $\beta \in$ N)
2. merge α, D

(10) simpler: **assign $\alpha \in$ N to D**

(11) First merge: Chomsky/Collins select 2 elements from N
Fortuny 2007 D = empty, (9) applies to first merge also

(12) D = series of stages. First stage D = zero.

D = $\{ D_0, D_1, D_2, \ldots , D_n \}$

nesting (Fortuny 2007): D_i contains D_{i-1}

D_i = output of Merge_i = assign α_i to D

(13) Merge

$M_i = f(\alpha_i) \rightarrow D_i$ where $\alpha \in$ N and $D_i = \langle \alpha_i, D_{i-1} \rangle$

(14) N = { John, Tense, loves, Mary }

$M_1 = f(\text{Mary}) \rightarrow D_1$ = Mary
$M_2 = f(\text{loves}) \rightarrow D_2$ = $\langle$ loves, Mary$\rangle$
$M_3 = f(\text{Tense}) \rightarrow D_3$ = $\langle$Tense, $\langle$ loves, Mary$\rangle\rangle$
$M_4 = f(\text{John}) \rightarrow D_4$ = $\langle$John, $\langle$Tense, $\langle$loves, Mary$\rangle\rangle\rangle$

(15) A derivation is a list of assignments from N to D

2. The LCA

(16) Hypothesis
the asymmetry between members of an ordered pair is relevant to the interfaces

(17)

A-P interface (PF):	prosody	(pitch accent)
	morphology	(dependency marking)
	linear order	(precedence)
C-I interface (LF):	interpretation	(predication/modification/scope interpretation)

(18) LCA: $\langle \alpha, \delta \rangle = / \alpha\ \delta /$

(19) ideally: dependency indicators at both interfaces *converge*

money schmoney	prosody	2nd member marked (money SCHMOney)
	morphology	2nd member reduplicates the 1st member
	semantics	2nd member functions as predicate of the 1st member

(20) domains of investigation:
juxtapositions, coordination, dependency marking (case, agreement), semantic dependencies (binding)

(21) NB linear order is the least reliable dependency indicator (movement)

3. Opacity (phases)

(22) Recall: N may contain any type of linguistic element

(23) A phrase is the output of a derivation (actually, applies to words as well)

(24) N contains a phrase only as the output of a previous derivation (**auxiliary derivation**)

(25) If T is a term of a phrase P included in a numeration N, T is not itself included in N (opacity)

(26) Movement: assign an element for the second time

(27) Hypothesis: an opaque domain is the output of an auxiliary derivation

(28) a. derives lexical integrity
b. derives the Condition on Extraction Domains (CED, Huang 1982), cf. Toyoshima 1997)
c. derives the Coordinate Structure Constraint (CSC, Ross 1967)

(29) Lexical intergrity

a.

manus-je	van	alles	
<name>-DIM	of	everything	'factotum, gopher'

b.

Hij	is	een	manusje van alles
he	is	a	factotum

c.

*Van alles	is	hij	een	manusje	
*Overal	is	hij	een	manusje van	(*overal* = everything)
*Een manusje	is	hij	van alles		

(30) a. N = { hij, is, een, [manusje van alles] }
b. N = { hij, is, een, manusje, van, alles } *would predict possibility of movement*

(31) CED (subject island)

a. Who did you see friends of ?
b. *Who did friends of see you ?

(32) a. N = {who, of, friends, see, did, you} (*friends of who* = complement)
b. N = {you, see, did, [friends of who] (*friends of who* = specifier)

a'. D = M1 who
M2 of who
M3 friends of who
M4 see friends of who
M5 you see friends of who
M6 did you see friends of who
M7 who you did see friends of who (*who* $\in$ N)

b'. D = M1 you
M2 see you
M3 [friends of who] see you (output of auxiliary derivation)
M4 did [friends of who] see you
M5 *who did [friends of who] see you (*who* $\notin$ N)

(33) ASIDE: a phrase constructed in D may be (re)assigned to D, e.g. from N in (31a):

a. Friends of who did you see ?

M7 [friends of who] did you see friends of who

This suggests that D and N are integrated, i.e. D is included in N

(34) Prediction: extraction from complement position always possible
extraction from specifier/adjunct position never possible
= CED

(35) Chomsky (2005: 13)
a. *It's the CAR of which [the driver --] caused a scandal
b. It's the CAR of which [the driver --] was arrested

(36) ASIDE: unpredicted (Chomsky 2005, 19)
a. It's the CAR of which [the driver --] is likely to cause a scandal
b. Of which car did they believe the driver -- to have caused a scandal

(37) (New) definition of **phase**
A phase is a complete derivation, where all members of N have been assigned, so:
A phase is a projection of N.

(cf. Chomsky: phase = vP/CP)

(38) Different predictions from Chomsky
a. no edge effects predicted (specifier of phase head as escape hatch)
b. CP need not be a phase

(39) wh-islands
a. Who do you think you saw ? (single N, no phases)
b. *Who did you wonder where you saw ? (unexplained)

(40) Movement from Spec,CP ? (Lasnik & Saito 1984)
a. Who said what ? (John said A, Bill said B, etc.)
b. Who said [what happened] ? (#John said A happened, Bill said B happened, etc)

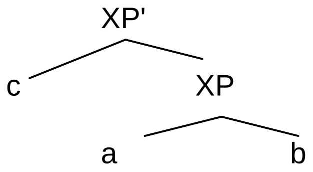
4. Recursion

(41) traditional view: nesting of identical categories

S → NP VP
VP → V S

(42) $[_S$ John said $[_S$ that Bill left]]
$[_{DP}$ the master of $[_{DP}$ the house]]

(43) minimalist view: XP within XP



(44) select a, b --> XP
select c, XP --> XP'

(45) Now: not so clear. (43) could be derived by iteration:

Merge b --> D = b
Merge a --> D = ⟨a, b⟩
Merge c --> D = ⟨c, ⟨a, b⟩⟩

(46) Even the nesting type (42) could be derived by iteration:

N = { the, master, of, the, house }

Merge *house* --> D = house
Merge *the* --> D = ⟨the, house⟩
etc.

(47) If merge = assign, there is much less recursion than previously thought

(48) Hypothesis:
recursion applies whenever the output of a derivation is included in N

(49) Recursion in language = derivational interaction, the ability to link derivations (phases)

(50) Every specifier or adjunct involves recursion (but subordination may not)

(51) Core case of recursion: coordination

5. Coordination / CSC

(52) Coordinate Structure Constraint
a. *Who do you love [John and --]
b. *I wonder who [John likes -- and Bill hates Mary]

(53) No extraction (out) of members of a coordinate structure (unexplained)

(54) noted exceptions

a. Across-The-Board I wonder who [John likes -- and Bill hates]
b. scene setting the whiskey I [went to the store and bought --]
c. contiguous the troops he wanted to [go and address --]
d. conative the thesis he wanted to [try and finish --]
e. such that ... 1 not the kind of guy you can [listen to -- and stay calm]
f. such that ... 2 the stuff those guys in the Caucasus [drink -- and live to be 100]

(55) Essentially two types:

a. complement type (54b,c,d): second member transparent
b. adjunct type (54e,f): first member transparent

(56) Complement type: second member is really a complement (Wiklund 2005)
--> first member = specifier (auxiliary derivation), hence a phase

(57) Adjunct type: second member is really an adjunct (Postal 1998) or conjunct (Kehler 2002)
--> output of aixiliary derivation, hence a phase

(58) Two types of CSC violations:

a. extraction of conjunct (52a)
b. extraction out of conjunct (52b)

(59) Logic now:

a. (52a): coordinate structure = output of auxiliary derivation (phase)
b. (52b): conjuncts are outputs of auxiliary derivations (phases)

(60) Intuitive difference subordination vs. coordination

a. subordination: [I know [that you know [that he knows [etc]]]
b. coordination: [[[A + B] + C] + D]

(61) To derive (60b):

N1 = { A, B } merge B --> D1 = B
merge A --> D1 = A + B

N2 = { A+B, C } merge C --> D2 = C
merge A+B --> D2 = A+B + C

N3 = { A+B+C, D } merge D --> D3 = D
merge A+B+C--> D3 = A+B+C + D

(62) Coordination

a. always a two-member N
b. no remerge (no movement)

(63) The grammar of coordination is more primitive (essentially juxtaposition)

(64) Arguably, 2-member N is the minimal N

(65) Chomsky (2005): numbers are derived by a 1-member N

merge 1 1
merge 1 2 etc

(66) Essentially, the numbers are derived not by merger/assignment, but by an operation **add 1**.

$M_i = \quad f(D_i) \rightarrow D_j \qquad \text{such that } D_j = D_{i+1} = D_i + 1$

(successor function)

(67) 2 is the minimal number of elements needed to generate structure

(68) Successor function + Lexicon (N) = ordered pairs = structure/information.

References

Ackema, Peter & Ad Neeleman. 2005. *Beyond Morphology*. Oxford University Press.

Bobaljik, Jonathan David. 1995. In terms of Merge. *MIT Working Papers in Linguistics* 27, 41-64.

Chomsky, Noam. 2005. On phases. Ms., MIT.

Collins, Chris. 1998. *Local economy*. MIT Press.

Fortuny Andreu, Jordi. 2007. The emergence of order in syntax. Dissertation, University of Barcelona.

Huang, C.T. James. 1982. *Logical relations in Chinese and the theory of grammar*. Dissertation, MIT.

Kehler, Andrew. 2002. *Coherence, reference, and the theory of grammar*. Stanford: CSLI Publications.

Lasnik, Howard and Mamoru Saito. 1984. On the nature of proper government. *Linguistic Inquiry* 15, 235-289.

Postal, Paul. 1998. *Three investigations of extraction*. MIT Press.

Ross, John Robert. 1967. *Constraints on variables in syntax*. Dissertation, MIT.

Stepanov, Arthur. 2001. The end of CED? *Proceedings of WCCFL 20*, 524-537.

Toyoshima, Takashi. 1997. Derivational CED. *Proceedings of WCCFL 15*, 505-519.

Wiklund, Anna Lena. 2005. *The syntax of tenselessness: on copying constructions in Swedish*. Dissertation, University of Umeå.

Faculty of Arts, PO Box 716, NL-9700 AS Groningen, The Netherlands
c.j.w.zwart@rug.nl • www.let.rug.nl/zwart/