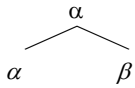
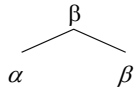
Symmetry in Syntax? Symmetric Merge and Symmetric Move

Barbara Citko (University of Washington)

While there has been a lot of research on asymmetry and antisymmetry in syntax, symmetry has been mostly ignored or claimed to be outright non-existent (Kayne 1994, Chomsky 1995, Moro 2000, Di Sciullo 2005). This focus on asymmetries and antisymmetries is somewhat surprising from a biolinguistic perspective, which seeks to integrate linguistics with the natural sciences (Boeckx and Piattelli-Palmarini (in press), Chomsky 2005, Jenkins 2000, among others), in which symmetry tends to be the norm and departures from symmetry are something that needs an explanation (Brody 2006). My main goal in this talk is to remedy this gap, by motivating the need for research on symmetry in syntax, and providing evidence for symmetric aspects of the two fundamental syntactic operations: Merge and Move. I will start by reviewing the arguments that led syntacticians to believe that Merge and Move are asymmetric.

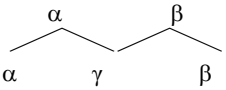
The asymmetry of Merge is twofold. First, given Kayne's (1994) Antisymmetry Theory, Merge can only create structures in which all terminal nodes are in asymmetric relationships, since only such structures can be linearized. This excludes structures with mutual c-command between terminal nodes, for example. Second, labeling, which is thought to be part of Merge, is asymmetric in that only one of the two merged elements projects as the label of the newly formed constituent (Chomsky 1995:244). For example, if two elements α and β merge, either α or β can project as the label, as shown in (1a-b).

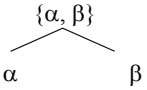
1) a. 

b. 

Various relativized minimality effects led us to believe that Move is also asymmetric. Well-known cases involve superiority effects, head movement constraint, and superraising. In all of them, only the higher of the two potentially movable elements may move.

In this talk, I show that, contrary to standard assumptions outlined above, both Merge and Move can be symmetric as well. I discuss cases two cases of symmetric Merge: *Parallel Merge* (Merge that creates structures with symmetric c-command relationships) and *Merge and Project Both* (Merge in which the labels of both α and β symmetrically project as the label of the new constituent). The two are illustrated in (2a) and (2b), respectively. Empirical evidence in favor of Parallel Merge comes from accross-the-board wh-questions (Citko 2005), conjoined wh wh-questions (Citko 2008, Gracanin-Yuksek 2007), serial verb constructions (Hiraiwa and Bodomo 2008), and right node raising constructions (Wilder 1999, Bachrach and Katzir to appear).

2) a. *Parallel Merge* 

b. *Merge and Project Both* 

Chomsky (1995) suggests *Merge and Project Both* is a theoretical possibility, as long as the two projecting elements do not conflict in relevant features. To the best of my knowledge, the empirical consequences of this theoretical possibility have not yet been investigated. I argue that this is what is involved in the derivation of comparative conditionals.

Next, I turn to cases of symmetric Move. Symmetric Move is Move that can target two elements with equally grammatical results. I discuss two cases of symmetric Move: passive movement in double object constructions and wh-movement in multiple wh-questions. In some languages, passivization is symmetric in that it can target either the direct or the indirect object, whereas in others it is asymmetric in that it only target one of them (Anagnostopoulou 2003, McGinnis 2001, Woolford 1993, Marantz 1993, Baker 1988, among many others). Wh-movement in multiple wh-questions exhibits similar variation; in some languages it is symmetric in that it target any wh-element in a clause (with no superiority violations), whereas in others it is asymmetric in that it has to target a higher one. The issue I conclude the talk with is the correlation (or lack thereof) between the availability of symmetric A movement and the availability of symmetric A-bar movement.