

# An Introduction to Nanosyntax via Classical Armenian Case Syncretism

John Gluckman. UCLA  
MoRG, April 15, 2015

April 15, 2015

## 1 What is Nanosyntax?

- Nanosyntax is a maximally decompositional theory of syntax, and by extension, morphology.

“[M]orphemes will span several terminals. And therefore they will correspond to an entire subtree rather than corresponding to a terminal.” (Starke, 2009, p. 2)

- The argument for Nanosyntax is that by considering the relationships between **features** as a hierarchical structure which is built using the same principles as syntax (binary branching Merge), then certain morphological patterns have a natural explanation.
- Nanosyntacticians have generally used paradigmatic syncretisms to argue their point. It’s essentially an argument for morphological blocking.
  - Certain forms are blocked because of hierarchical **structure**.
  - There is no need to rely on an independent morphological module.
- There are three fundamental principles/assumptions.
  - (1) *Bottom-up spell-out*  
Spell-out proceeds bottom up and material can be “over-written” by a later spell-out rule.

(2) *The Superset Principle (simple version)*

A phonological exponent is inserted into a node if its lexical entry contains all features of that node.

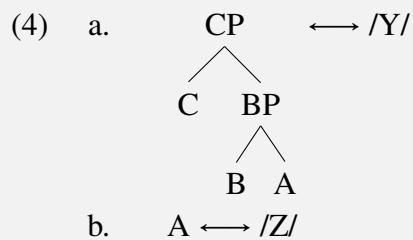
(3) *The Elsewhere Condition*

In case two rules (R1 and R2) can apply in an environment E, R1 takes precedence over R2 if it applies in a proper subset of environments compared to R2.

- Instead of feature bundles, phonological forms are direct mappings between structures and sounds.<sup>1</sup>

## A sample derivation

Some rules for spell-out



- Let's say the syntax builds this phrase marker without a CP,  $BP$



- Working from the **bottom up**, A has the possible realizations of /Y/ and /Z/.
- B has the possible realizations of /Y/
- Thus, BP is realized as /Y/, via **set intersection**

- The key result is that because of the Superset Principle, the Elsewhere Condition, and syntactic structure, certain syncretisms become impossible to derive.<sup>2</sup>

<sup>1</sup>The same mapping is presumed to exist between structure and meaning. This will not be discussed here.

<sup>2</sup>Technically this isn't quite true. There is always accidental homophony. As every theory is going to need to allow this, I don't think it should be held against Nanosyntax.

## Comparing the Superset and Subset Principles

**Subset Principle**

- If [A,B,C] corresponds to an exponent /X/, /X/ will be inserted for any bundle that comes out of the syntax containing [A,B,C,(D,...)].
- **The lexical information is a (possibly non-proper) subset of what is generated in the derivation.**

**Superset Principle**

- If [A,B,C] corresponds to an exponent /X/, /X/ will be inserted for any bundle (=structure) that comes out of the syntax which contains A, B, or C and nothing else.
- **The lexical information is (possibly non-proper) superset of what is generated in the derivation.**

## 2 Armenian case syncretisms

- Caha is attempting to account for the empirical generalization about Classical Armenian in (5).

(5) *Case Contiguity (Armenian)*

- Non-accidental case syncretism is restricted to contiguous regions in the following sequence:
- NOM – ACC – LOC – GEN – DAT – ABL – INS

- Contiguity is a generalization about which syncretisms we *don't* see.<sup>3</sup>

– We don't see, e.g., NOM/LOC syncretisms, *where ACC is not also syncretic*.<sup>4</sup>

<sup>3</sup>Clearly by reordering the hierarchy of cases, we'd lose the generalization. But Caha supports such a hierarchy based on a cross-linguistic evidence.

<sup>4</sup>Caha does not address 'year.LOC.SG' versus 'year.ABL.SG, nor the fact the -ê shows up in some ABL forms and not others. I return to this later.



## Evidence for case features from Case Attraction

- Caha shows that some languages display “case attraction”, where a dependent nominal assumes the case of the nominal it’s dependent on, i.e., in possessive structures. (Case Attraction is optional in Armenian.)
- (9) a. *i knoj-ê t’agawor-I-n*  
by wife-ABL king-GEN.SG-DEF  
‘by the wife of the king’
- b. *i knoj-ê t’agawor-Ê-n*  
by wife-ABL king-ABL.SG-DEF  
‘by the wife of the king’
- (10) Case attraction as agreement ellipsis  
head noun-case<sub>*i*</sub> [[dependent noun-GEN] AGR=case<sub>*i*</sub>]
- Caha analyzes this as copying of the possessee’s case to the possessor, and then ellipsis applies to the case features of the possessor.
  - Caha argues for the ellipsis account because of the “recoverability” of ellipsis, noting that such Case Attraction only happens with cases that are of equal or greater feature content as genitive.

## An ellipsis account of the facts

	N-GEN-AGR	Ellipsis	Output
a.	N-GEN-NOM	N-[A,B,C,D]-[A]	*
b.	N-GEN-ACC	N-[A,B,C,D]-[A,B]	*
c.	N-GEN-LOC	N-[A,B,C,D]-[A,B,C]	*
d.	N-GEN-GEN	N-[A,B,C,D]-[A,B,C,D]	N-GEN
e.	N-GEN-DAT	N-[A,B,C,D]-[A,B,C,D,E]	N-DAT
f.	N-GEN-ABL	N-[A,B,C,D]-[A,B,C,D,E,F]	N-ABL
g.	N-GEN-INS	N-[A,B,C,D]-[A,B,C,D,E,F,G]	N-INS

- Assuming that this decomposition is correct, then Caha can straightforwardly derive Case Contiguity.
- For instance, consider the following supposedly unattested Case paradigm.

An offending paradigm

case	form
NOM	$\alpha$
ACC	$\beta$
LOC	$\alpha$

- To get NOM and LOC to be syncretic, we need something like (11).

$$(11) [C,B,A] \longleftrightarrow / \alpha /$$

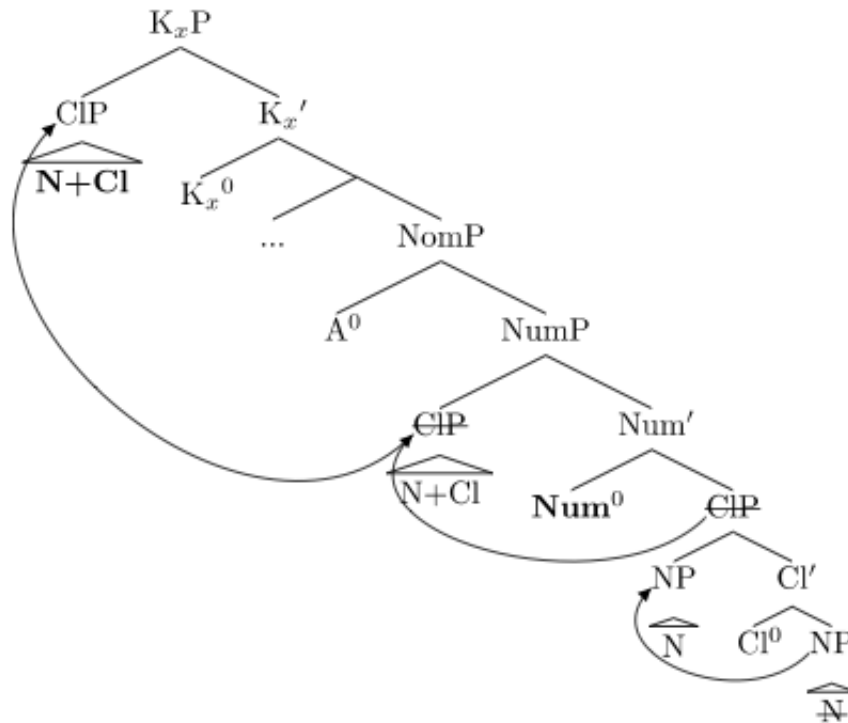
- Note that because these are syntactic structures, it would be impossible to leave [B] out.
- Now we need an entry which will spell out the features [A, B], because these are the accusative features.<sup>6</sup>

$$(12) [B,A] \longleftrightarrow / \beta /$$

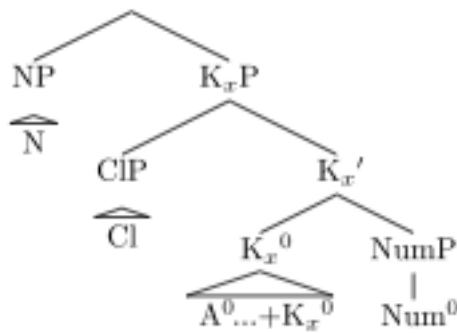
- Crucially, though, NOM is going to apply to (12) as well, because A is listed in [B,A], and moreover, it's a more restrictive environment, so we would expect NOM to end up being  $/ \beta /$ .
- Thus, the  $\alpha \beta \alpha$  pattern is impossible to create.<sup>7</sup>
- In this system, only constituents can be spelled-out as one morpheme, thus he needs a fairly fancy syntactic derivation.
  - Note that the tree here leaves out the last step of sub-extraction of the NP out of CIP, so that CIP forms a constituent with the other phrases.
  - Furthermore, all the Case heads undergo head-movement onto to K.

<sup>6</sup>We couldn't have just [B] spell-out as  $/ \beta /$ , as this would then block being able to spell-out LOC as just  $\alpha$ . This should become clearer in a minute.

<sup>7</sup>See also Bobaljik (2013)'s discussion of \*ABA in comparative morphology.



- Because all traces are ignored during spell-out, the structure can be represented as the simpler one below, which is the input for phonological insertion.



- Because of his assumptions about how spell-out works, Caha makes the following prediction.

(13) *Agglutinative vs. fusional spell-out (Armenian):*

- a. In the sequence below, if a given case is expressed together with some other category, than all cases to its left are as well
- b. NOM – ACC – LOC – GEN – DAT – ABL – INS

***The template for Armenian declension***

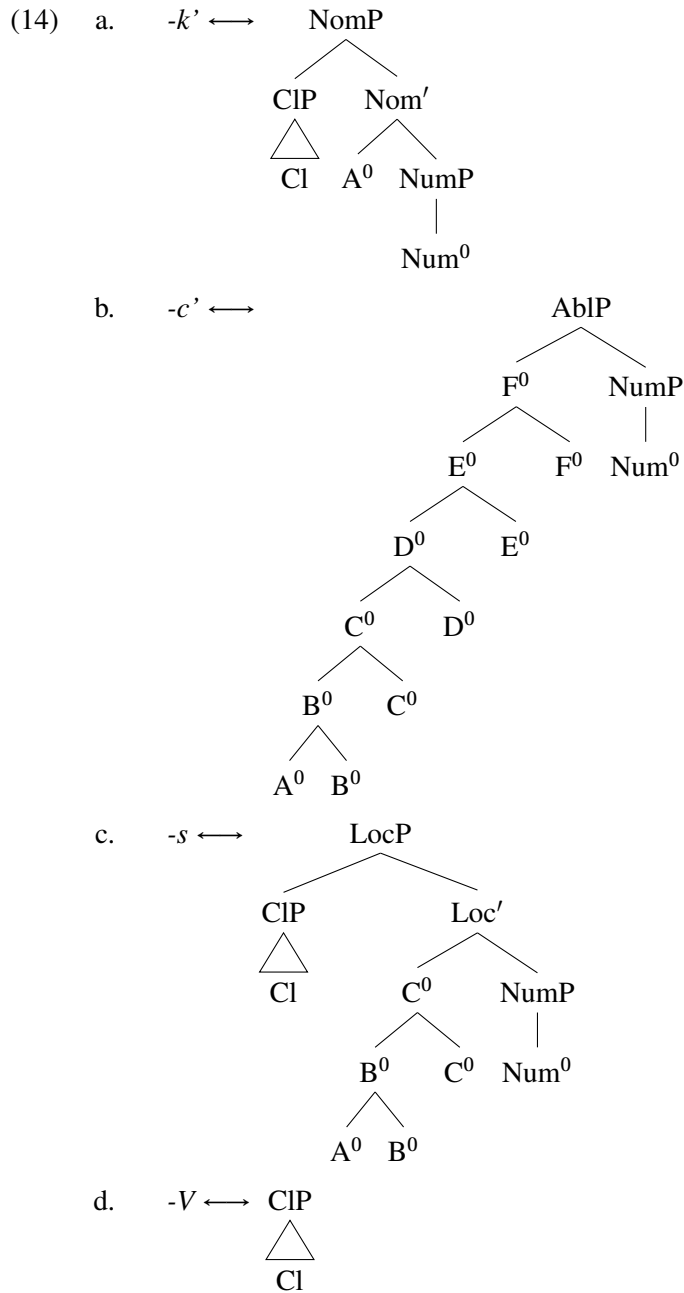
	STEM	CLASS	CASE	NUMBER
NOM, ACC, LOC	stem	-class&case&number		
GEN, DAT, ABL	stem	-class	-case&number	
INS	stem	-class	-case	-number

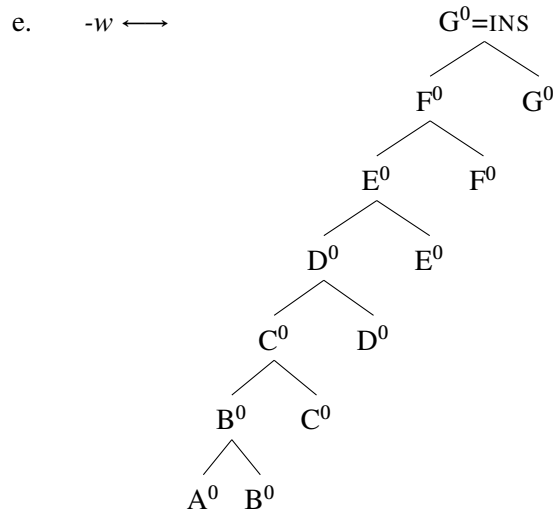
***Case, Number and Class***

	STEM	CLASS	CASE	NUMBER
NOM	stem		-k'	
ACC, LOC	stem		-s	
GEN, DAT, ABL	stem	-a-/-o-/-i-/-u-		-c'
INS	stem	-a-/-o-/-i-/-u-	-w-	-k

- We're now set to start deriving the Armenian patterns.
- The following are Caha's specifications for phonological insertion.







- How to read these correspondences.
  - The lettered branching structures result from head movement of case features. Since traces are ignored, they are left out of the structures for convenience.
  - I think we’re supposed to assume that Num<sup>0</sup> bears [PL] for the derivations below.
- It should be clear that the majority of the work in this system is done in these correspondences, which are extremely free.
  - For instance,  $-w$  spells-out a complex head and is a “pure” case morpheme, while  $-s$  is a phrase, and captures class, case, and number.

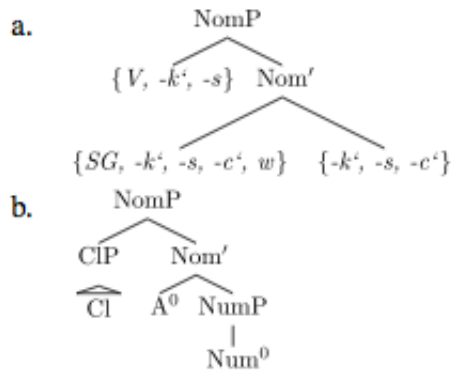
(15) *The Superset Principle (revised)*

A lexically stored tree matches a syntactic node iff

- a. the lexically stored tree contains the syntactic node including the material dominated by that node OR
- b. the lexically stored tree matches all daughter nodes (ignoring traces).

## 1. Nominative

- We work from the bottom up, replacing each node with its possible phonological form, according to the Superset principle.<sup>8</sup>



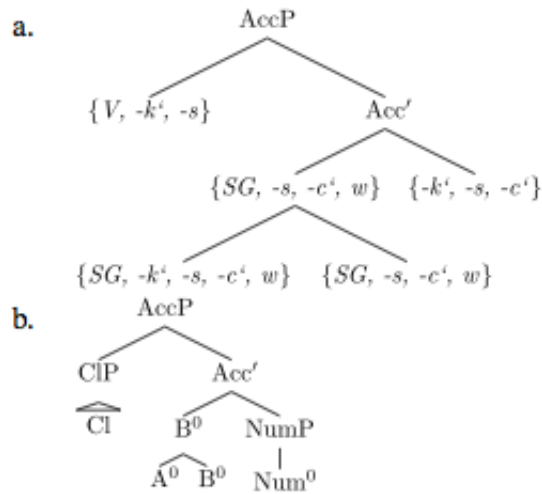
- NumP can be realized as  $-k'$ ,  $-s$ , or  $-c'$ , while  $A^0$  can be  $SG$ ,  $-k$ ,  $-s$ ,  $c'$ , or  $-w$ .<sup>9</sup>
- Set intersection yields  $Nom' = \{-k', -s, -c'\}$ .
- Set intersection with CIP yields  $\{-k', -s\}$ .
- The Elsewhere Condition applies, and  $-k'$  wins, because it applies in a more restrictive environment.

## 2. Accusative

- With accusative, the process is essentially the same, except that  $-k'$  is eliminated early, and so  $-s$  will win.

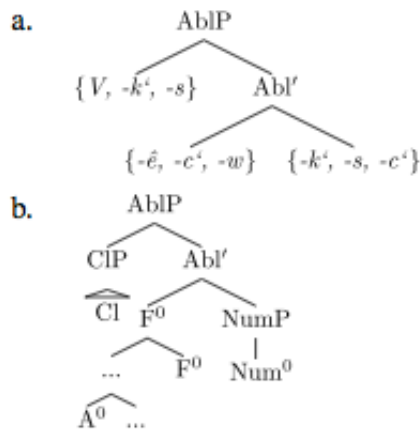
<sup>8</sup>Note that we can “override” spell-out as we precede up the tree. This is called CYCLIC OVERRIDE, which is also the name of my new band.

<sup>9</sup>I frankly cannot figure out why  $A^0$  has the option to be  $SG$ .



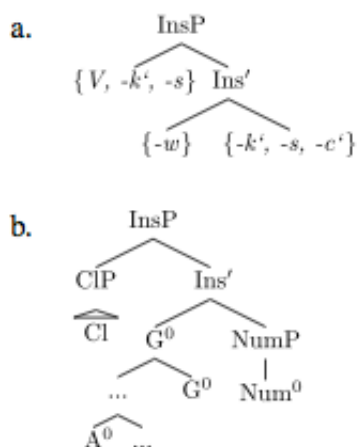
### 3. Ablative

- The ablative is interesting because Abl' and CIP do not share any features in common, and so here we will see two morphemes spelled-out.



### 4. Instrumental

- And the instrumental is further decomposed for the exact same reason.



- So once we adopt some of the Nanosyntax assumptions, derivations proceed fairly cleanly.
  - The main assumption is the dependence on a certain syntactic configuration.

### 3 Issues

- I think there are a number of places where we could try to tear apart Caha's system here, but that's not really fair to him. Any morphological theory is going to have complications, and we need to grant Caha some leeway.
  - That said, there are some non-trivial issues that should be addressed.
1. The case pattern of *tari-*, 'year', seems to be an exception to Caha's system.

#### *Attested syncretisms in Classical Armenian*

	word (SG.)	nation (PL.)	nation (SG.)	year (SG.)	river (PL.)
NOM	bay-∅	azg-k'	azg-∅	tari-∅	get-k'
ACC	bay-∅	azg-s	azg-∅	tari-∅	get-s
LOC	bay-i	azg-s	azg-i	tarw-oj̄	get-s
GEN	bay-i	azg-ac'	azg-i	tarw-oy	get-oc'
DAT	bay-i	azg-ac'	azg-i	tarw-oy	get-oc'
ABL	bay-ê	azg-ac'	azg-ê	tarw-oj̄ê	get-oc'
INS	bay-iw	azg-awk'	azg-aw	tare-aw	get-owk'

2. Does this account explain patterns *within the entire paradigm*?
  - Singular and plural seem to have a template. This is essentially accidental under this system.
  - Moreover, it's not clear to me if he has a way to handle the differences between noun classes.
  - Again, *tari-* is going to need special rules.
    - Note that I'm not objecting to specifying the noun-classes; this is going to have to be a part of every system.
    - Rather, I'm worried about the fact that, granted the specification, there are still patterns that match across the singular paradigm.
  
3. If there is ever allomorphy of the noun based on case (and the case features are preserved) the system will fail.
  - That is, if there were some stem *chun*, that irregularly became *chuvac'* in the genitive plural, then Caha would not be able to handle this.
  - I don't know the answer to this, but I suspect that at least some nouns have irregular forms for some cases. And if not in Armenian, then surely some other language.
  
4. Is there room for phonologically conditioned allomorphy?
  - Rephrased: **How** does phonologically triggered allomorphy take place?
    - Starke (2009) claims that Nanosyntax offers a solution to mora-based templatic restrictions. (e.g., some roots need two moras) but he doesn't explain it's not obvious to me.
  - Caha sort of addresses this question with respect to what looks to me like epenthesis to fix a bad consonant structure, which he analyses by adding another projection into the structure.
  
- Overall, this is an interesting proposal, that seems to be internally consistent, and thus I think a theory worth investigating.
  - If this approach is adopted, it requires a very specific view of syntax (i.e., a strongly Kaynean one).
  - It further radically "repurposes" the Lexicon, which no longer now just consists of discrete features, which are assembled to make the words, phrases, morphemes in the syntax.

## References

- Bobaljik, J. D. (2013). *Universals in Comparative Morphology*. MIT press.
- Caha, P. (2013). Explaining the structure of case paradigms by the mechanisms of Nanosyntax. *Natural Language & Linguistics Theory*, 31:1015–1066.
- Starke, M. (2009). Nanosyntax: A short primer to a new approach to language. *Nordlyd, special issue on Nanosyntax*, 36(1):1–6.
- Taraldsen, K. T. (2009). The nanosyntax of Nguni noun class prefixes and concords. *Lingua*, 120:1522–1548.