Coordination and Neutralization in HPSG

Roger Levy and Carl Pollard

Stanford University and Ohio State University

1 Introduction

Ingria (1990) pointed out that neutralization (also called indeterminacy), including the special case of feature value syncretism, seemed to pose a problem for unification-based grammar. Few linguists adopted Ingria's proposal, which involved replacing unification in the relevant cases with a nondistinctness check, but the paper was influential in the sense that it persuaded many people that unification-based grammars and the constraint-based frameworks that superseded them were a hopeless case. In the mid-90's, when Bayer and Johnson (1995) showed how to solve the problems in question using Lambek's additive meet type constructor, it made a big impact and helped persuade many people to abandon constraint-based grammar for type-logical grammar.

In recent years type-logical accounts of coordination of unlikes have been closely linked to those of neutralization. On the approaches of Bayer and Johnson, and Heylen (1996), which build on earlier work by Morrill (1990, 1994), coordination is treated in terms of Lambek's additive join constructor, so that coordination and neutralization are lattice-theoretic dual operations (where the lattice in question is the Lindenbaum algebra of the Lambek calculus, namely the free residuated lattice generated by the basic type symbols). In the prosodic interpretation ("frame semantics"), these operations correspond, respectively, to intersection and union of sets of word strings.

In this paper, we use lattice theory to present an account of neutralization and coordination of unlikes that can easily be incorporated into current HPSG, where linguistic objects are modelled as totally well-typed, sort resolved feature structures. The account, which is consistent with the HPSG analysis of neutralization proposed by Levine et al. (2001), was developed by the alphabetically first author during autumn 2000, and the lattice-theoretic presentation by the alphabetically second author during winter and spring 2001. Another HPSG-based account of the same phenomena, developed independently by Daniels (2001) at about the same time, is shown by lattice-theoretic methods to be essentially equivalent to ours. In both our approach and Daniels', neutralization and coordination turn out to be dual lattice operations, just as in the type-logical account; the lattices in question are, however, quite different from the type-logical one. Unfortunately, space limitations necessitate postponing detailed critique of the type-logical account, which we intend to provide elsewhere. The same applies to the recent LFG-based account of Dalrymple and Kaplan (2000), which employs set values in several different ways to model neutrality, coordination, ambiguity, multiple selection (e.g. of adjuncts), and 'principled resolution' of agreement features.

Proceedings of the 8th International HPSG Conference, Norwegian University of Science and Technology (3-5 August 2001)
Frank van Eynde, Lars Hellan and Dorothee Beermann (editors)
2002.CSLI Publications. http://csli-publications.stanford.edu/

2 Neutralization, with Syncretism as a Special Case

Neutralization is occasioned when one and the same sign appears in a context where it appears to be involved in the satisfaction of mutually inconsistent constraints. For the purposes of this paper, we divide cases of neutrality into two main kinds: ARGUMENT neutralization–also known as syncretism–and FUNCTOR neutralization.

Argument neutralization consists of cases where a single sign satisfies seemingly inconsistent valence requirements. Familiar examples are given in (1):

(1) Argument neutralization (more specifically, case syncretism)

a. Er findet und hilft FRAUEN/*MÄNNER/*KINDERN. (German; coordination)

b. Kogo Janek lubi a Jerzy nienawidzi? (Polish; coordination)
who.ACC/GEN like.ACC and hate.GEN

c. Kogo/*Čego/*Čto ja iskal, ne bylo doma. (Russian; free
who.ACC/GEN / *what.GEN/*what.ACC I sought, not was home
relative)

"What I was looking for wasn't at home."

d. This is someone WHO/*WHOM even close friends of __ believe __ must be watched like a hawk. (parasitic gap)

Argument neutralization has to be distinguished from argument ambiguity, illustrated in (2):

(2) Argument ambiguity: failure of ambiguous signs to permit argument neutralization

a. *The sheep that is ready are here.

b. *Co Janek zrobil a zmartwilo Marie? (Polish)
what.ACC/GEN did and upset

c. *Sie singt und singen. (German)
SHE/THEY sings and sing

On the other hand, functor neutralization, also called neutralizable polyvalency, is occasioned when a single unsaturated sign has seemingly mutually inconsistent valence requirements simultaneously satisfied by two different signs, namely a coordination of unlikes, as in (3):

(3) Functor neutralization (neutralizable polyvalency)

a. Kim is a Republican and proud of it.

b. Včera vec' den' on prozhdal svoju podrugu Irinu i zvonka
Yesterday all day he expected self's.ACC girlfriend.ACC Irina.ACC and call.GEN
ot svoego brata Grigorija. (Russian)
self's brother Gregory

"Yesterday he waited all day for his girlfriend Irina and for a call from his brother Gregory."

Functor neutralization has to be sharply distinguished from functor ambiguity, also called non-neutralizable polyvalency, illustrated in (4):

(4) Functor ambiguity (non-neutralizable polyvalency): failure to permit functor neutralization

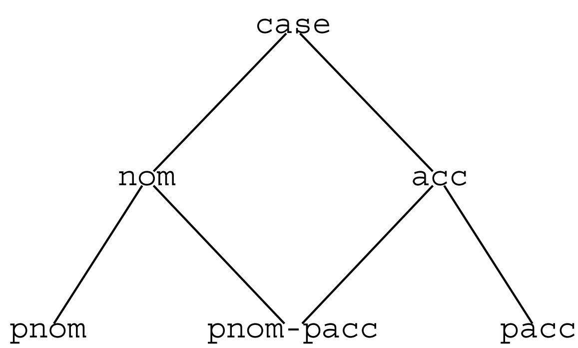
a. *You can tuna and say that again.

b. *Sandy waxed his car and melancholic.

Note that we have not exhausted the space of logically possible neutralizations. For example, languages such as Russian and German show different case syncretism patterns on adjectives than on nouns, and we might imagine that an adjective syncretized for nominative and accusative cases could appear modifying a coordinate noun consisting of a nominative and an accusative conjunct. Empirically, however, this never seems to happen; we refer the reader to Levy (2001) for further discussion.

The first HPSG analysis of case syncretism, as in (1), appeared in Levine et al (2001), in the context of parasitic gaps in English. For languages like English that have nominative and accusative and allow them to syncretize, as in (1d), Levine et al. proposed a type hierarchy of case values like the one in (5).

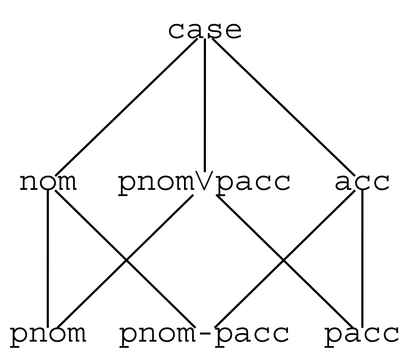
(5) Levine et al.'s type hierarchy for a two-case system



Here `pnom` ("pure nominative") is the case value of a nominative pronoun like *she*, and nom-acc is the value for a syncretic form like *who*. The value nom is essentially the logical disjunction of `pnom` and `nom-acc`, and that is what a finite verb selects for the case of its subject, since either a pure nominative or a nom-acc syncretic can be a subject.

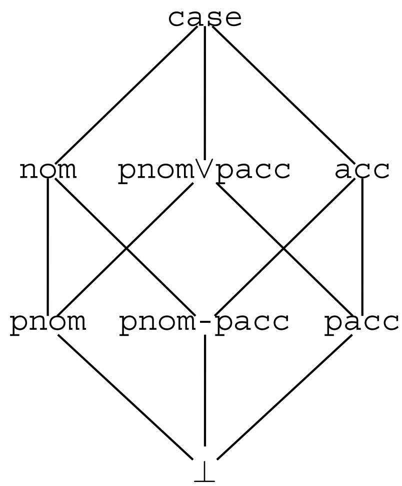
One question left open is how Levine et al.'s approach can be generalized to the arbitrary case of syncretisms of n basic feature values. It starts to become clearer when we fill in the missing node in (5):

(6) Levine et al.'s type hierarchy for a two-case system with missing node added



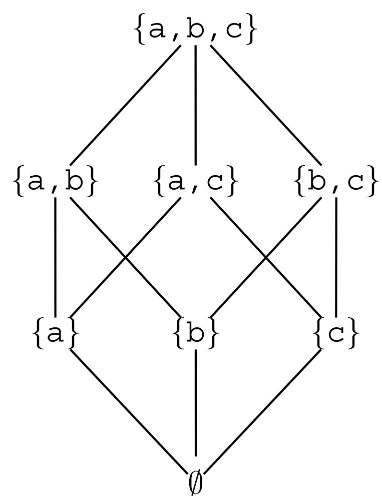
This missing node is the logical disjunction of pnom and pacc. It's no wonder Levine et al. omitted it from their account, since it is hard to imagine a language with nom-acc syncretism having a verb that would accept either a pure nominative or a pure accusative, but not a syncretic form. Note that this hierarchy forms a join semilattice with seven nodes. That may seem a bit arcane, but now consider the same picture with another node added at the bottom for the empty type:

(7) Levine et al.'s type hierarchy for a two-case system with missing node and bottom added



Perhaps this picture looks a little more familiar: it is a boolean algebra isomorphic to the power set of a three-element set, as shown in (8):

(8) A boolean algebra (the powerset of {a,b,c})



On this perspective, the Levine et al. type hierarchy for a two-case system is a join semilattice isomorphic to the semilattice obtained by taking the powerset of a three-element set and deleting the empty set–three because three is one less than the the cardinality of the powerset of {pnom, pacc}. In general, the Levine et al. type hierarchy extended in this way to a set S of basic elements will be isomorphic to

(9) $Pow(Pow(S) - \emptyset) - \emptyset$

The motivation here is that syncretized case values can be seen as additional maximally specific elements of the *case* hierarchy, corresponding to singleton sets in the powerset lattice, and the ability of case governors like verbs to take either pure or syncretized elements arises from the fact that their value for some selecting feature occupies a position in the case hierarchy which is not maximally specific (or equivalently, which is disjunctive), corresponding to nonempty nonsingleton sets in the powerset algebra. So in a three-case system, this version of the Levine et al. hierarchy would be isomorphic to the semilattice obtained by taking the powerset of a 7-element set and tossing out the empty set, giving a total of 127 nodes. In a four-case system, the Levine et al. hierarchy would have 32,767 nodes.

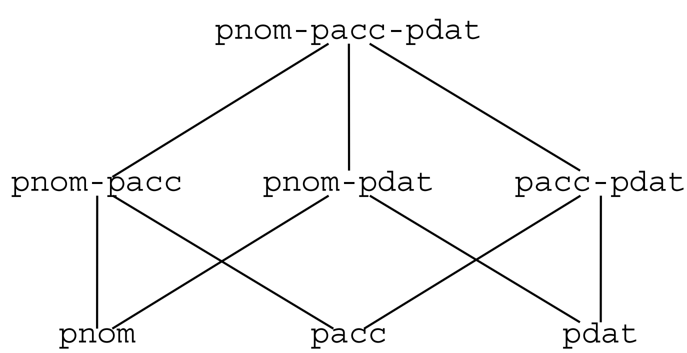
Now let us examine the situation in a system with three pure case values. This case is complex enough to be instructive, while yielding a lattice small enough to depict graphically. Including the three pure values, which we arbitrarily call pnom,pacc,pdat, and their possible syncretizations, there are seven resulting atomic case values:

(10) The seven atomic case values in a three-case syncretization system

```
pnom pacc pdat pnom-pacc
pnom-pdat pacc-pdat pnom-pacc-pdat
```

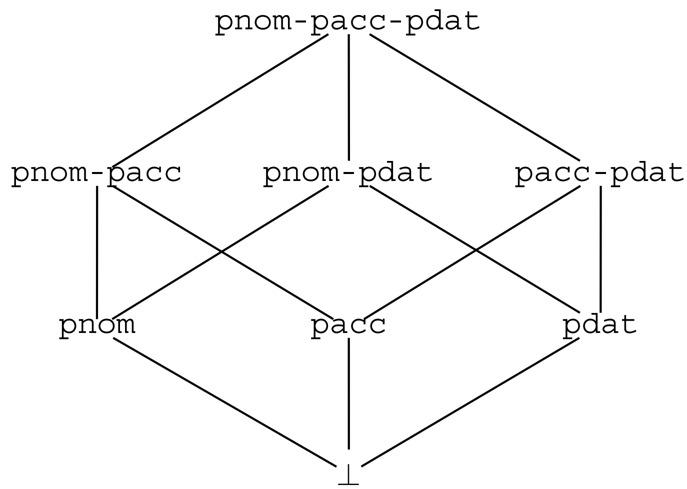
Note that in the approach to syncretization that we examined above, these atomic case values were mutually unordered. There is an inherent ordering of syncretized case values, though, which we can see if we take the hyphen representing syncretization in (10) as a semilattice join operation. This yields the following:

(11) The syncretization semilattice for a three-case system



This semilattice is isomorphic to the one in (6), but its origin is completely different. Correspondingly, the intepretation of the join operation is completely different: it represents not logical disjunction, but rather syncretization. In fact, we can turn the semilattice in (11) into a boolean algebra by throwing in a bottom element, though it is linguistically meaningless because there could never be a sign that syncretized an empty set of pure case values. That is shown in (12):

(12) The syncretization boolean algebra for a three-case system (atoms correspond to Levine et al.'s pure values)



The picture in (12) is the essence of all existing formal analyses of case syncretization that we are familiar with. In Levy (2001), syncretization is a lattice operation; noun case values, syncretized or unsyncretized, specify a position on the lattice, and selecting values specify a lower bound on the lattice. This is isomorphic to the treatment in Dalrymple and Kaplan (2000), where cases are set-valued: we identify the bottom row of three elements with their singleton sets and the hyphen as set union. Finally, if we interpret the node labels as types in categorial grammar and the hyphen as Lambek's additive meet, then we obtain the analysis of syncretism in Bayer (1996).

3 Coordination

We take Wasow's generalization (Pullum and Zwicky, 1986) as the empirical generalization to be accounted for:

(13) Wasow's Generalization, paraphrased: Leaving aside cases of "principled resolution" (Corbett, 1983), the conjuncts of a coordinate structure are subject to whatever constraints apply to the coordinate structure as a whole. (cf. the property distribution schema in Bresnan et al. (1985)).

The traditional motivating evidence for Wasow's generalization has been coordination of constituents of unlike syntactic category, as in the coordination of NP and AP objects in (3a), repeated here:

(3a) Kim is a Republican and proud of it.

This example of functor neutralization can actually be handled within the neutralization hierarchy presented in the previous section, by reversing the positions of argument values and functors' selecting values. Whereas for case syncretism, as we said, arguments' values specified positions on the lattice in (12), and verbs' selecting values specified lower bounds, for coordination we would take the verb's selecting value as a specified position, and the arguments' syntactic categories as lower bounds. If in (12), we replaced all occurrences of `pnom` with NP and `pacc` with AP, then the resulting sublattice is a picture of unlike-category constituent coordination.

But notice that this is a unidirectional analysis: given a syntactic feature that shows neutralization on *one of* arguments or functors, we construct an isomorphic powerset lattice and put the unneutralized features on the second row from the bottom, then use the nodes above to represent neutralizations. As a result, there is no room to do both argument *and* functor neutralization on the same feature. A claim that powerset lattices are a complete picture of neutralizations would be a claim that given a syntactic feature in a given language, functors *or* arguments may be neutralized, but never both.

There is at least one class of counterexamples to this: case in Slavic, where the well-known accusative-genitive neutralization extends not only to argument neutralization, as in (1c), but also to functor neutralization, as in (3b):

(1c) Kogo/*Čego/*Čto ja iskal, ne bylo doma. (Russian; free relative)
who.ACC/GEN / *what.GEN/*what.ACC I sought, not was home

"What I was looking for wasn't at home."

(3b) Včera vec' den' on prozhdal svoju podrugu Irinu i zvonka ot svoego brata Grigorija. (Russian)
Yesterday all day he expected self's.ACC girlfriend.ACC Irina.ACC and call.GEN self's brother Gregory

"Yesterday he waited all day for his girlfriend Irina and for a call from his brother Gregory."

Given that we can already account for both argument and functor coordination, one at a time, on lattices, the natural question is, what is the lattice on which we can account for them both?

Our approach to answering this question takes as its point of departure the following principle due to Bayer (1996):

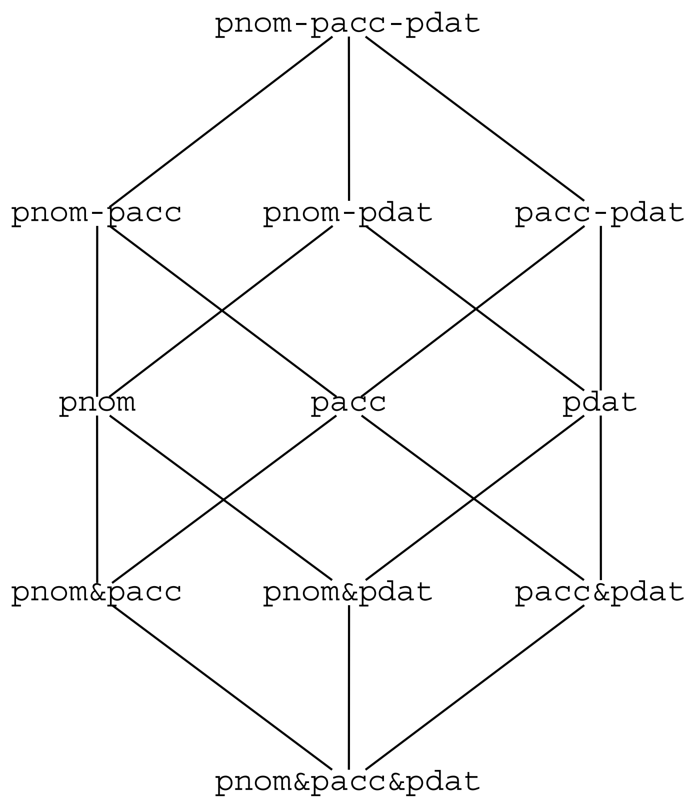
(14) Bayer's Principle: Coordination and neutralization are dual lattice operations in a distributive lattice.

As noted above, in the type-logical treatment, the lattice in question is a lattice of sets of word strings (that is, a sublattice of the powerset of a free monoid), with coordination and neutralization corresponding to set union and set intersection respectively (these being the respective prosodic interpretations of Lambek's additive join and additive meet type constructors); it must be distributive because it is a sublattice of a powerset lattice. Of course, we cannot use the same lattice, because HPSG is not a form of type logical grammar: feature values (including syntactic categories, which are values of the path SYNSEM|LOC|CAT) are not interpreted as sets of word strings. Instead our lattice will be an abstract mathematical object constructed out of "basic" feature values.

First, why concretely *can't* we simply take the lattice in question to be the powerset lattice (12)? That is: if the join in this lattice is syncretization, why not just take the meet to be coordination? For example, why couldn't the value `pnom` represent the case value for a coordinate NP whose conjuncts are a nom-acc syncretic and a nom-dat syncretic? The problem with this idea is that it runs afoul of Wasow's generalization, since such a coordinate structure could also be the object of a verb that could take either an accusative or a dative, but a nominative NP could not. Introducing the notation & for the meet (coordination) operation, the problem in question amounts to failure to distinguish nom from (nom-acc)&(nom-dat).

A second attempt might be to drop the useless bottom element off the powerset lattice and reflect the result along its bottom row of three elements. This would give the following lattice:

(15) First attempt to enlarge (12) into a lattice where syncretization (-) is join and coordination (&) is meet



This lattice, however, has exactly the same defect: it can't distinguish nom from (nom-acc)&(nom-dat).

So far we have clarified one desirable property of the sought syncretization-coordination lattice:

(16) The syncretization-coordination lattice must be large enough to distingish between (say) pnom and (pnom-pacc)&(pnom-pdat).

Now let us touch on another:

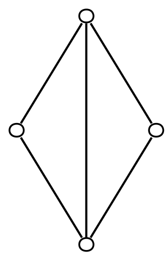
(17) The syncretization-coordination lattice should not be so large that it distinguishes between (say) pnom-pacc and (pnom-pacc)&(pnom-pacc-pdat).

This is because there is no point proliferating the number of possible syncretization-coordination feature values beyond what can be distinguished by the sets of selecting environments that they can occur in. Thus if two feature values can appear in exactly the same selecting environments, then

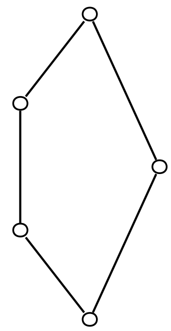
Occam's Razor compels us to regard them as the same value. In (17), the `(pnom-pacc)` conjunct in `(pnom-pacc)&(pnom-pacc-pdat)` eliminates the possibility of the coordinate structure being selected by a verb that governs dative; therefore `(pnom-pacc)` and `(pnom-pacc)&(pnom-pacc-pdat)` can appear in exactly the same selecting environments, and they should not be distinguished in the syncretization-coordination lattice.

Following Levy (2001), we propose that the lattice in Figure 1 is the right syncretization-coordination lattice for three pure case values.[1] Join represents syncretization, and meet is coordination. From inspection we can see that this lattice is self-dual, and that it is distributive since a lattice is known to be distributive iff it does not contain any sublattices of the forms shown in (18).

(18) The lattice in Figure 1 is distributive, since it contains neither



nor



as a sublattice.

These facts do not depend on CASE having three pure values, but extend to the corresponding lattices for syncretizable features with any number of pure values, as we will show in the next section.

As explained in the previous section, a case-syncretized argument will have a case value specified by the syncretization, for example *kogo* in Russian, syncretized between accusative and genitive:

(19) Argument Neutralization:
An argument that is neutral between accusative and genitive cases is specified as
`[CASE pacc-pgen]`

Given this lattice, what does it mean for a predicate to govern, say, dative case? It means that the verb can take as object any NP whose case value is either `pdat` or else syncretizes `pdat` with one or more other pure case values. In terms of the lattice in Figure 1, we can describe this set of options as the set of all elements bounded below by `pdat`; that is, it is the principal filter generated by `pdat`, written `^pdat`. This is shown below.

(20) Argument Selection:

a. The NPs that can serve as objects of predicates that govern (say) dative case are the ones whose case values lie in `^pdat`, the principal filter generated by `pdat`, where
`^pdat = {pdat, pnom-pdat, pacc-pdat, pnom-pacc-pdat`
`(pnom-pdat)&(pacc-pdat)}`

[1] In the presentation in Levy 2001, there are actually extra topmost and bottommost elements, but they play no essential role in the analysis we present here, and we omit them.

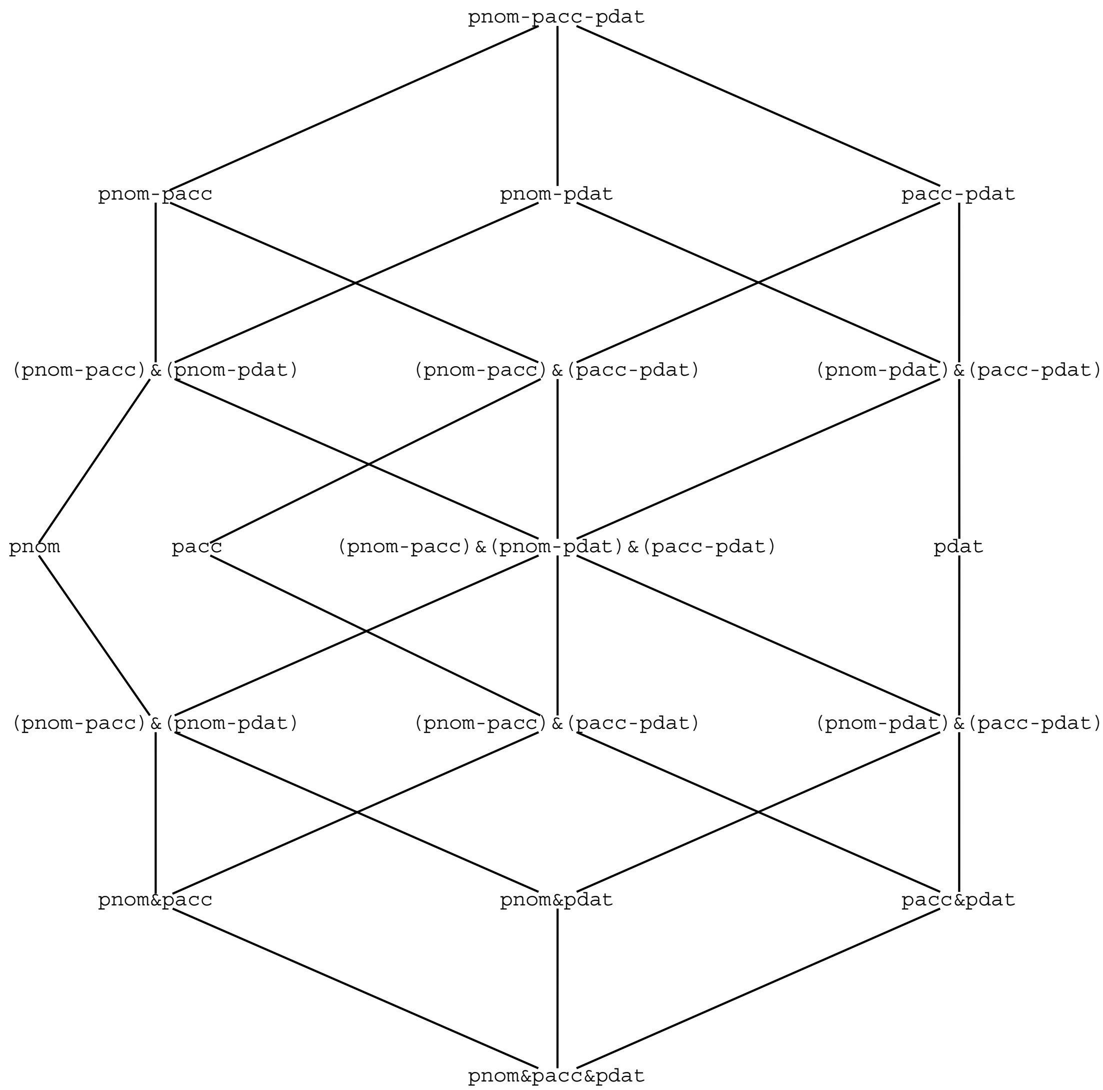


Figure 1: Syncretization-coordination lattice for three pure case values.

b. Therefore if a dative-governing verb is described as
[COMPS <NP[dat]>]
the value dat must be defined as the logical disjunction of all the values in ^pdat, i.e. dat means $\bigvee$^pdat, i.e.
pdat $\vee$ pnom-pdat $\vee$ pacc-pdat $\vee$ pnom-pacc-pdat
$\vee$ (pnom-pdat)&(pacc-pdat)

c. This is a slight refinement of how Levine et al. would define dat, which would contain all these disjuncts except the last one (the one permitting the verb to have a coordinate object whose conjuncts are a nominative-dative syncretic and an accusative-dative syncretic).

In the case of functor neutralization, governing case values are still disjunctions over principal filters, but now the filters in question are generated by neutralized values. For the situation in (3b), where the verb can take as an object the coordination of accusative and genitive NPs, we would have the following:[2]

(21) Functor neutralization:
A neutralizably polyvalent predicate that is neutral between governing accusative and genitive is specified as
[COMPS <NP[CASE $\bigvee$^(pacc&pgen)]>]
i.e. the case specification for the complement is the disjunction of the all the values in the principal filter generated by pacc&pgen.

Thus argument neutralization crucially involves the syncretization operation, functor neutralization crucially involves the coordination operation. This reflects a deep connection between duality in the syncretization-coordination lattice and the distinction between head features and valence features.

By contrast with functor neutralization, a verb that is merely ambiguous between governing accusative and governing genitive would be specified as
[COMPS <NP[acc $\vee$ gen]>]
i.e. the case specification for the complement is the disjunction of all the values in the union of the two principal filters generated by pacc and pgen respectively:
acc $\vee$ gen = $\vee$(^pacc $\cup$ ^pgen)
Crucially, this set of disjuncts contains pacc-pgen but not pacc&pgen.

4 Generalized Syncretization and Coordination

The remainder of this paper shows how the construction of the syncretization-coordination lattice, of which Figure 1 is the case for three pure values, can be generalized using the Smyth powerlattice, adapted from a domain-theoretic concept employed used in programming language semantics.

[2]For purposes of discussion, assume pgen to be substituted for pdat in (1). In Russian, which has six cases, the actual lattice of syncretizations and coordinations will be much larger; but the three-case (and the two-case) lattices will be sublattices, so the analysis holds.

The main ideas of the construction are as follows. A lattice similar to the one in Figure 1 (but with n pure values, for any positive integer n) is obtained by starting with a boolean algebra B like the one in (12) (but with n atoms), forming its Smyth powerlattice (defined below), and discarding its top and bottom elements. Any lattice so constructed can be shown to be self-dual and distributive (in fact, it is the free DeMorgan lattice on n generators, but we will will not need this fact). In addition, such a lattice contains as a sub-join-semilattice the join-semilattice obtained from the boolean algebra B by removing its bottom element (for the case $n = 3$, this is the semilattice in (11)). Thus embedding the syncretization semilattice into the syncretization-coordination lattice preserves the syncretization operation; from this it follows that the Levine et al. treatment of neutralization carries over to our extended account which also includes coordination.

First, some definitions:

(22) Definition of upper-closed (and lower closed) subset: Given a lattice (or more generally, a partially ordered set (poset)) L,

a. a subset S of L is UPPER-CLOSED iff for every p in S and every q in L ordered above p, q is also in S.

b. a subset S of L is LOWER-CLOSED iff for every p in S and every q in L ordered below p, q is also in S.

c. Obvious fact: S is upper-closed iff its complement L-S is lower-closed.

d. Obvious facts: The union and intersection of two upper-closed (lower-closed) subsets are also upper-closed (lower-closed).

(23) Definition of the Smyth powerlattice of a lattice (or poset)
For a lattice (or poset) L, Smyth(L), the Smyth powerlattice of L is defined as follows:

a. The members of Smyth(L) are the upper-closed subsets of L;

b. If S and T are two members of Smyth(L), then S is ordered below T in Smyth(L) iff T is a proper subset of S.

c. The join in Smyth(L) of S and T is their intersection.

d. The meet in Smyth(L) of S and T is their union.

(24) Since Smyth(L) is a sublattice of the lattice dual to the powerset lattice Pow(L), it is therefore distributive.

To get more insight into the Smyth powerlattice, we need the concept of *irredundant subsets*:

(25) Irredundant subsets:

a. A subset S of a lattice (or poset)) L is IRREDUNDANT if its elements are pairwise incomparable, i.e. no element of S is above another relative to the ordering on L.

b. Example: if S is the lattice in (12), {`pnom-pacc, pdat`} is irredundant, but {`pnom-pacc, pacc`} is not.

c. Singleton subsets of L are obviously irredundant.

d. The set Min(S) of minimal elements (relative to the ordering on L) of S is irredundant.

(26) Theorem. Let L be a poset which is finite (or more generally, well-founded, i.e. it has no infinite descending chains), S a member of Smyth(L), and M = Min(S). Then S is the union of the principal filters generated by the members of M.
Suppose moreover that T is another member of Smyth(L) with N = Min(T). Then

a. S is ordered below T in Smyth(L) iff every member of N is bounded below in L by a member of M.
b. the meet of S and T in Smyth(L) is the union of the principal filters generated by the members of Min(M U N).
c. Suppose moreover that L is a join-semilattice with join operation - . Then the join of S and T in Smyth(L) is the union of the principal filters generated by the members of Min(Q), where Q is the subset of L consisting of all elements of the form p-q with p in Min(S) and q in Min(T).
d. Under the same hypotheses, the function h: L $\rightarrow$ Smyth(L) that sends each p in L to its principal filter ˆp is a join-semilattice embedding, i.e. it is one-to-one, monotonic, and for all p, q, h(p-q) is the join in Smyth(L) of h(p) and h(q).

(27) Corollary. Let L be a finite lattice (or, more generally, a well-founded join-semilattice). Then Smyth(L) is isomorphic to the lattice smyth(L) defined as follows, with the isomorphism mapping each S in Smyth(L) to its set of minimal elements:

a. The members of smyth(L) are the irredundant subsets of L, and for any two members A and B of smyth(L):
b. A is ordered below B in smyth(L) iff for every member of B there is a member of A that is ordered below it in L.
c. The join of A and B in in smyth(L) is Min(C), where C is the subset of L consisting of all elements of the form p-q with p in A and q in B.
d. The meet of A and B in smyth(L) is Min(A U B).

From the foregoing, the claims stated at the beginning of this section follow straightforwardly. To begin with, Levy 2001 defines the lattice in Figure 1 in such a way that it is transparently the same as smyth(L) where L is the boolean algebra in (12), minus its top and bottom elements (which would correspond respectively to the linguistically meaningless neutralization and coordination of the empty set of pure case values).

By (24), Smyth(L) is distributive; discarding its top and bottom elements doesn't change that. And by (27), the lattice in Figure 1 must also be distributive.

The foregoing facts together with (26d) show that the syncretization semilattice in (11), which is the lattice in (12) with its bottom removed, embeds as a join-semilattice into the lattice in Figure 1.

And the self-duality of the lattice in Figure 1 is an immediate consequence of the fact that Smyth(L) is self-dual whenever L is. That in turn follows easily from the facts that:

(28) for two upper-closed subsets of L, S and T, S is a subset of T iff L-T is a subset of L-S; and

(29) S is upper-closed in L iff L-S is upper-closed in L under the dual ordering.

In closing, we note that the lattice proposed by Daniels (2001) (his (22)) can be seen to be precisely Smyth(L), from which it follows that his lattice and ours are isomorphic.

References

Bayer, S. (1996). The coordination of unlike categories. *Language*, 72(3):579–616.

Bayer, S. and Johnson, M. (1995). Features and agreement. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 70–76. Association for Computational Linguistics.

Bresnan, J., Kaplan, R. M., and Peterson, P. (1985). Coordination and the flow of information through phrase structure. MS, Xerox PARC.

Corbett, G. (1983). Resolution rules: Agreement in person, number, and gender. In Gazdar, G., Klein, E., and Pullum, G. K., editors, *Order, Concord, and Constituency*. Dordrecht, Holland; Cinnaminson, N.J.: Foris.

Dalrymple, M. and Kaplan, R. (2000). Feature indeterminacy and feature resolution. *Language*, 76(4):759–798.

Daniels, M. (2001). On a type-based analysis of feature neutrality and the coordination of unlikes. In *Proceedings of the 7th International Conference on Head-Driven Phrase Structure Grammar*.

Heylen, D. (1996). On the proper use of booleans in categorial logic. In *Formal Grammar*, pages 71–84. Prague.

Ingria, R. J. P. (1990). The limits of unification. In *Proceedings of the 28th Annual Meeting of the ACL*, pages 194–204. Association for Computational Lingusitics.

Levine, R. D., Hukari, T. E., and Calcagno, M. (2001). Parasitic gaps in English: Some overlooked cases and their theoretical implications. In Culicover, P. W. and Postal, P. M., editors, *Parasitic Gaps*, pages 181–222. MIT Press.

Levy, R. (2001). Feature indeterminacy and the coordination of unlikes in a totally well-typed HPSG. MS, Stanford University.

Morrill, G. (1990). Grammar and logical types. In Stokhof, M. and Torenvliet, L., editors, *Proceedings of the 1989 Seventh Amsterdam Colloquium*, pages 429–450. Also in G. Barry and G. Morrill 1990 (eds.), *Studies in Categorial Grammar*, Edinburgh Working Papers in Cognitive Science, Volume 5, 127-148. Revised version appears as Morrill 1996: *Grammar and Logic*, in *Theoria*, *Vol. LXII*, *Part 3*, pp. 260-293.

Morrill, G. V. (1994). *Type Logical Grammar: Categorial Logic of Signs*. Kluwer.

Pullum, G. K. and Zwicky, A. M. (1986). Phonological resolution of syntactic feature conflict. *Language*, 62(4):751–773.