DISTRIBUTION OF PHI-FEATURES WITHIN DPS AND THE ACTIVITY CONDITION

SAKUMI INOKUMA
Jissen Women’s University

This article proposes a theoretical bridge between DP-internal syntax and clausal syntax. A bundle of φ-features, composed of three sub-features [Person], [Number], and [Gender], is supposed to behave as an atomic, non-decomposable element in clausal syntax, while recent studies show that inside a DP, these sub-features are scattered and occur on different heads. We need a precise mechanism that “stacks up” the scattered sub-features and makes them available to clausal syntax as a single φ-bundle. I argue that a radical revision of the theory of feature-inheritance advocated by Chomsky (2007, 2008) and Richards (2007) provides a theoretical solution to this problem.*

Keywords: phases, probe-goal Agree, feature-inheritance, structured phi-features

1. Introduction

In investigating the nature of nominal elements, we need a clear definition of the category at issue. That is to say, we first have to have a theory as explicit and principled as possible that explains the core, prototypical aspects of the nominal category with regard to its semantic, morphological, phonological and/or computational properties. Otherwise we would end up with impressionistic and stipulative discussion. Focusing on the computational aspects, our theory must be mechanically well-defined on the basis of fundamental entities and basic relations such as formal features and probe-goal Agree. Many recent works tend to embrace either of the two theoretical concerns: toward highly abstract phase-based models, or finely grained, cartography-oriented models. In many cases, however, they remain silent

* The analysis presented in this article is a totally revised and extended version of Inokuma (2007). I am grateful to Noriko Imanishi, Akira Watanabe, and anonymous EL reviewers for valuable comments and suggestions. I have also benefited from the fruitful discussion with Shun’ichiro Inada, Yuki Ishihara and Chigusa Morita in preparing for this project. Remaining inadequacies are solely mine.
as to how to reconcile these two models.\(^1\)

With this perspective in mind, this article points out several theoretical gaps, which have always been left unanswered in spite of their robustness, at the interface of nominal structure and clausal structure. The problems revolve around the treatment of φ-features inside and outside a nominal phrase, which interacts with questions about Case features. The problems are simple. As reviewed in the following section, a φ-feature has traditionally been treated as if it were an “atomic” element in clausal syntax, while accumulating insights on nominal syntax in the last twenty years or so have made it clear that, within a nominal phrase, sub-features of a φ-feature, that is, [Person], [Number] and [Gender], are distributed on distinct heads. What has not been made clear yet is how we can make sure that these distributed sub-features are “stacked up” for a clausal domain to treat them as a single unit.\(^2\)

To fill in this theoretical gap is the goal of the present article. For that purpose, theoretical parsimony is given the first priority. I will start by examining existing theories and evaluating their theoretical and logical validity, and go on to point out another permissible course of derivation.

The article is organized as follows. Section 2 clarifies the theoretical issues by reviewing the standard treatment of φ-features with regard to probe-goal Agree (Chomsky (2000) et seq.) and their decomposition in nominal syntax. Basic theoretical tools made use of in the present work are also summarized. Section 3 is devoted to the presentation and reevaluation of the theory of feature-inheritance put forth by Chomsky (2007, 2008) and Richards (2007). The discussion in Section 3.1 will make it clear that Richards’s (2007) reduction of feature-inheritance to the Phase Impenetrability Condition (PIC) dismisses certain logically permissible paths for a derivation to converge. Combining the results of Sections 2 and 3.1, Section 3.2 enters into technical details of nominal syntax and argues that the paths dismissed by Richards (2007) and Chomsky (2007, 2008) are indeed employed in this domain. Section 4 concludes.

\(^1\) Chomsky (2008: 143), for instance, states that “[…] C is shorthand for the region that Rizzi (1997) calls the “left periphery,” possibly involving feature spread from fewer functional heads (maybe only one).” As it stands, it is not clear what it means to be “shorthand” for a category, nor what mechanism lies behind the “feature spread.”

\(^2\) Recently several works are concerned with (some aspects of) the theoretical problems discussed here, for example Danon (2011), Pesetsky and Torrego (2007), and Carstens (2012), among others. Time and space do not allow me to fully review these works, though.
2. Dual Status of Phi-Features: Atomic in Clauses, Scattered in Nominal Phrases

In standard theories of $\phi$-feature agreement (Chomsky (1995, 2000, 2001)), $\phi$-features on a nominal phrase are treated as a single unit, or a bundle of sub-features [Person], [Number] and [Gender].

(1) *The clausal side: $\phi$-feature as a bundle*
   a. An uninterpretable $\phi$-feature on C-T or $v^*$ agrees with an interpretable $\phi$-feature on DP.
   b. A $\phi$-feature consists of person, number and gender.

If any one of the sub-features is absent on the nominal phrase, agreement fails. This is stated as $\phi$-Completeness Condition in (2).

(2) *$\phi$-Completeness Condition:*
   For agreement to be successful (and for valuation to take place), a $\phi$-feature must be complete.

This condition is used by Chomsky (2000, 2001) to explain the derivation of the *there*-type expletive construction in English.

(3) a. There exist/*exists several theoretical problems.
   b. There *exist/exists an easy way out.

According to Chomsky (2000, 2001), expletive *there* is $\phi$-incomplete, in that it has only [Person]; hence agreement between *there* and finite T, for example, does not succeed. The T then goes on to search for an appropriate nominal phrase with a full $\phi$-set. If it finds one, the uninterpretable $\phi$-feature on T is successfully valued and deleted. Notice along the way that the $\phi$-Completeness Condition also applies for clausal heads, in a fashion parallel to nominal phrases (4). Agreement fails with Raising/ECM-type T, whose $\phi$-features are assumed to be incomplete, again only with [Person] (Chomsky (2001: 7)).

(4) a. *It seems Tom to be tired.
   b. Tom seems to be tired.

When we turn to the nominal-internal domain, however, recent approaches show that each of these sub-features originates on a separate designated head (5). If so, and if we are to stick to the $\phi$-Completeness Condition, then we must make sure that the sub-features distributed within a DP are

\[3\] I will not go into detailed discussion of the ultimate analyses of each feature. The point relevant to our concern is only that, when we look into nominal structure, $\phi$-features are scattered and far from “atomic.”
“stacked up” on a single head. Danon (2011) calls this (often tacit) assumption “the DP encapsulation assumption.”

(5) The nominal side: distributed φ-features
   a. Person features originate on D⁰.  
      (Bernstein (2008), Longobardi (2008))
   b. Number features originate on Num⁰.  (Ritter (1991))
   c. Gender/Class features originate on Class⁰ or N⁰.  
      (Picallo (1991), Ritter (1993))

(6) The DP Encapsulation Assumption  (Danon (2011: 299))
   In every nonexpletive noun phrase, the highest head bearing φ-features is φ-complete.  
   Now we have a puzzle to solve: how and why are the sub-features of a φ-set on a single nominal phrase, which are distributed within the phrase, “stacked up” and made available to clausal heads as a single unit?

   In investigating the above question, I take theoretical parsimony as the first principle. By this I mean that, especially in a context where purely theoretical problems are of concern, we must avoid any additional technical devices and assumptions as far as possible. So let us begin by enumerating the assumptions that we are going to employ. In addition to the locus of sub-features as given in (5), the Probe-Goal Agree relation is adopted, along with the Activity Condition on the goal, as in (7).

(7) a. Probe-Goal Agree:
   An uninterpretable feature on a head probes the interpretable counterpart on the goal.
   b. Activity Condition on (φ-)Agree:
      The goal must have an uninterpretable feature to be visible for Agree.

   One thing to notice at this point with regard to the Activity Condition is that the condition in effect requires uninterpretable features (uFs) both on the probe and on the goal for the system of Agree to work. In other words, in the probe-goal relation, uFs function “upward” on the goal (External Agree) as well as “downward” on the probe (Internal Agree).⁴

⁴ Carstens (2012) takes up the same “upward” effect of uF on the goal and discusses how to reduce it to the standard, “downward” uF, along the line of Bošković (2007). This article takes a different path. See below.
(8) Both the probe and the goal must have uninterpretable features for Agree to succeed.
   i. For the probe, uninterpretable features work “downward.”
      \[ \text{[INTERNAL AGREE]} \]
   ii. For the goal, uninterpretable features work “upward.”
      \[ \text{[EXTERNAL AGREE]} \]

With these assumptions, let us consider how the Agree relation is established in clauses and in nominals in turn. In the clausal domain, a classical view is given in (9), where C-T and v* carry an uninterpretable \( \phi \)-feature ([u\( \phi \)]), and the argument DP carries its interpretable counterpart, plus uninterpretable Case features ([uCase]), which makes the DP active as a goal.

(9) The Clausal Side:
   a. The probe (on C-T and v*) has [u\( \phi \)].
   b. The goal (on DP) has [\( \phi \)] and [uCase].

When we turn to the nominal domain, things become more complicated. If we tentatively adopt the DP-structure below (recall (5)), the first question is how to state the \( \phi \)-Completeness Condition.

(10)
\[
\begin{array}{c}
\text{DP} \\
D_{[\text{Per}]} \\
\text{Num}_{[\text{Num}]} \\
N_{[\text{Gend}]} \ldots
\end{array}
\]

Since the sub-features of the \( \phi \)-feature on DP are split over distinct heads, if nothing happens inside this DP, no single head could serve as the \( \phi \)-complete goal for the clausal \( \phi \)-probe. One possibility might be to abandon the \( \phi \)-Completeness Condition altogether and to allow each sub-feature to enter into an independent Agree relation. This seems unlikely, however, since it forces us to give up the fundamental correspondence between \( \phi \)-completeness and argument-licensing. Defective agreement phenomena such as in the expletive there construction and the Raising/ECM construction would lose an explanation, too.\(^5\)

The above consideration leads us to a fairly conservative analysis: namely, taking \( D^0 \) as the locus of the complete \( \phi \)-set. Let us adopt this as a core assumption and formulate it as in (11).

\(^5\) See also Danon (2011).
(11) At the stage of $\phi$-probing by a clausal head, $D^0$ must be $\phi$-complete.
In order for $D^0$ to be $\phi$-complete, it must obtain (the values for) [Num] and [Gend] features in the course of derivation. A natural mechanism for attaining this result, in a parsimonious computational model, is probe-goal Agree. Given the definition in (7a), we expect $D^0$ to have uninterpretable [Num] and [Gend], as schematized in (12).

Given the DP structure in (12), the uninterpretable features [uNum] and [uGend] probe down the internal domain of the $D^0$, in an ordinary fashion.$^6$ However, a computational problem arises with regard to the treatment of the sub-features on $D^0$ thus valued within its DP domain. Unlike the Agree relation found in the clausal domain, the valued uninterpretable sub-features, namely [uNum] and [uGend], must remain in the derivation. If [uNum] and [uGend] were deleted at this point of the derivation, $D^0$ would end up with only [Per], and the $\phi$-completeness problem would come up again. We will come back to this problem in the following section, and show that no additional mechanisms are needed to overcome the problem.

Finally, to satisfy the Activity Condition (9b), the (argument) DP must carry [uCase], which means in the present perspective that the $D^0$ has the relevant feature, since a full suite of the $\phi$-set at stake can only be found on $D^0$. The state of affairs is summarized in (13).

---

$^6$ In fact, as an anonymous EL reviewer points out, this DP-internal probing by [uNum] and [uGend] already raises questions to the probe-goal system outlined here. The questions are whether the Activity Condition as stated in (7b) is operative in this case, and if so, how. These questions are of high importance in the present context, where nothing should be added to the theory unless necessary. For theoretical consistency, the present article tentatively assumes that the answer to the first question is positive, and that the uninterpretable features required on the goals ($N^0$ and $N^0$) are [uGend] and [uNum], respectively (cf. Heycock and Zamparelli (2005), Harbour (2011)). The assumption is empirically supported by DP-internal agreement (“concord”) phenomena with respect to Number and Gender, but not Person (cf. Baker (2008)). For some inconclusive discussion, see Section 4.
(13) *The Nominal Side:*

a. $D^0$ must carry $[\text{uNum}]$ and $[\text{uGend}]$, which probe downward inside DP.

b. Thus valued $[\text{uNum}]$ and $[\text{uGend}]$ must be available to operations outside DP.

c. $D^0$ must carry $[\text{uCase}]$; furthermore, this $[\text{uCase}]$ does not probe downward.

In light of the standard probe-goal Agree system, (13b) and (13c) seem problematic. According to (13b), $[\text{uNum}]$ and $[\text{uGend}]$ survive in the derivation after they are valued via DP-internal Agree, to be available to operations outside, that is, the operations in the clausal domain. According to (13c), on the other hand, $[\text{uCase}]$ on $D^0$ does not probe downwards inside the DP, but rather serves as an activator of the $D^0$ for the sake of “external agreement” (recall (8ii)). These two mechanical problems lurk in any approaches to the nominal-clausal interface, quite independently of their empirical coverage. In the next section, I will argue that these two theoretical problems can be resolved in a unified fashion, by revising the theory of feature-inheritance advocated by Chomsky (2007, 2008) and Richards (2007).

3. Feature-(Non-)Inheritance

In this section we will look at how the two theoretical problems pointed out in (13b, c) can be explained on principled grounds by modifying the theory of feature-inheritance. After reviewing the logic of feature-inheritance theory by Richards (2007) and Chomsky (2007, 2008), we will proceed to modification of the theory in Section 3.2.


Chomsky (2008) suggests that all syntactic operations are driven solely by phase heads. What this implies is that T is neither a phase head nor a probe. The operations apparently driven by T (e.g. $\theta$-Agree and A-movement into SpecT) are in fact triggered by the phase head C, which comes immediately above T. Uninterpretable features brought into the derivation by a phase head (i.e. C) percolate down to the lower non-phase head (i.e. T), and they probe from this “derived head” position. Chomsky (2007) calls this mechanism *feature-inheritance.*

---

7 Putting aside External Merge that builds up the structure (Chomsky (2007: 17)).
Although Chomsky’s (2007) argument is based on conceptual consideration of language design, i.e. the rigid “derivation-by-phase” model and the identification of probes with phase heads C and v*, Richards (2007) provides a stronger argument for the feature-inheritance theory, in a sense that feature-inheritance is automatically derived by the Phase Impenetrability Condition (PIC).8

(14) Phase Impenetrability Condition (PIC):
The internal domain of a phase is inaccessible to any operations outside of it.

Richards’s (2007) argument goes as follows. Following Chomsky (2000, 2001, 2008), uninterpretable features are unvalued in the lexicon, and these features must be valued and deleted in the course of (narrow) syntax, in accordance with Full Interpretation (FI); otherwise they cause crash at the interfaces. The crucial assumption here is that the computational system does not distinguish valued uninterpretable features from interpretable features. Hence, valuation and deletion (as part of Transfer) must happen “simultaneously.” We thus obtain (15) as an interface condition.

(15) Premise 1:
Value and Transfer of uFs must happen together.  
(Richards (2007: 566))

Second, again following Chomsky (2000, 2001, 2008), Richards (2007) assumes that what is transferred to the interfaces is the complement domain of phase heads: TP for a C phase, VP for a v* phase. This is natural when we consider that something must remain in the narrow syntax in order to feed further computation; otherwise no structural relations could be defined between different phases.9

(16) Premise 2:
The edge and nonedge (complement) of a phase are transferred separately.  
(Richards (2007: 568))

From these two premises, Richards (2007) concludes that feature-inheritance necessarily holds as in (17), since otherwise all the derivations would result in automatic crash at every phase level.

---

8 On a side note, the theory of feature-inheritance originates in the manuscript of Chomsky (2008), circulated as Chomsky (2005), and Richards (2007) refers to this prepublication version. Chomsky (2007), which was written as a sequel to Chomsky (2005), in turn adopts Richards’s (2007) arguments.

9 See Fox and Pesetsky (2005) and Fukui and Kasai (2005) for a different conception for a spell-out domain, where the concern is mainly with the phonological interface.
(17) **Conclusion:**

uF must spread from edge to nonedge (i.e. from C to T, v* to V, etc).

(Richards (2007: 569))

Richards (2007) calls this deduction the *feature-inheritance corollary* and takes the conclusion in (17), which is correctly derived from the combination of premises (15) and (16), as the only logical possibility of avoiding automatic crash. However, we have other paths to avoid automatic crash while keeping to premises (15) and (16). In the next subsection I will argue that Richards’s corollary is invalid and that allowing uFs to be uninherit- ed brings about interesting consequences for DP-internal syntax.

3.2. Feature-Non-Inheritance and the Treatment of Features inside and outside DP

Recall the logic of Richards’s (2007) argument. His corollary is summarized in (18).

(18) **Feature-Inheritance Corollary (Richards (2007)):**

a. Unvalued uF on a non-Phase-Head (non-PH) causes automatic crash at that phase.

b. Valued uF on a non-PH is successfully deleted at that phase.

c. Unvalued uF on a PH causes automatic crash at the next phase.

d. Valued uF on a PH causes automatic crash at the next phase.

According to Richards (2007), the problem arises when a uF is valued through the probe-goal Agree relation with a phase head PH1 in a phase P1, and yet it remains untransferred in P1 to enter the next phase P2. The higher phase head PH2 can never “know” that the valued F in question is in fact an uninterpretable feature valued in the preceding phase P1, and treats it as an interpretable feature. There is no way to successfully delete this feature, resulting in crash at the semantic (i.e. CI) interface at P2 level. Thus any uninterpretable features brought into the computational system by phase heads must be inherited by non-phase heads, which are inside the complement (i.e. internal) domain of its phase head.

This line of argumentation does rationalize the validity of the feature-inheritance mechanism, but does not necessitate the mechanism. If, as Richards (2007) claims, allowing valuation and transfer of a certain uninterpretable feature to take place in different phases causes crash, then we could try the opposite strategy of Richards’s (2007): leave the feature for the next phase; no inheritance, no valuation. As long as the feature stays in the phase head, there is no problem in transferring its internal domain to the
interfaces.

To make this discussion concrete, let us go through all the combinatorial possibilities of two operations: inheritance and valuation. First, consider (18a), schematized in (19). Here we have inheritance and transfer, but not valuation. The uninterpretable feature on PH₁ is inherited by X, rendering PH₁ “clean.” But the inherited feature is not valued within the domain of PH₁ and is transferred without a value. This unvalued feature causes crash at the P₁ level.

(19) *Inherit + Transfer:

\[
\text{PH}_1 \xrightarrow{[\text{uF}]} X \xrightarrow{[\text{uF}]} \ldots \alpha \ldots \xrightarrow{\text{TRANSFER}}
\]

The second case (18b), schematized in (20), is the only configuration that Richards (2007) argues is allowed in the computational system. Non-phase head X inherits uninterpretable features on PH₁, making PH₁ clean. Then this derived probe X Agrees and values those features. Transfer applies to the internal domain of PH₁, in which uninterpretable features inherited by X are successfully valued and are subject to deletion.

(20) \(\checkmark\text{Inherit} + \text{Value} + \text{Transfer}:

\[
\text{PH}_1 \xrightarrow{[\text{uF}]} X \xrightarrow{[\text{uF}]} \ldots \alpha \ldots \xrightarrow{\text{TRANSFER}}
\]

The two cases above are unproblematic and I have nothing to add to Richards’s (2007) and Chomsky’s (2007, 2008) analysis.

The third case, which I would like to argue is possible under the model developed by Chomsky (2007, 2008) and Richards (2007), is given in (21). In this structure, nothing other than Transfer happens. PH₁ carries uninterpretable feature(s) into the derivation, but neither inheritance nor valuation occurs. Transfer applies as usual, leaving the PH₁ (and its uninterpretable features) in narrow syntax. At this level, there is no problem with regard to Full Interpretation at the interfaces and PIC.
(21) **Transfer Only:**

The two premises of Richards (2007) are satisfied “vacuously” in (21). Since no problem arises in P₁ phase in (21), we must check whether this orphan uF causes automatic crash at the next phase. If it does, (21) must be excluded on principled grounds; if it does not, we expect some empirical support for this option.

Finally, the most intriguing case would be (18d), schematized in (22). In this configuration, no inheritance occurs and the uninterpretable feature is valued “in situ” on the phase head PH₁. Since Transfer sends only the internal domain of a phase, the valued uninterpretable feature on PH₁ is left untransferred.

(22) **Value + Transfer:**

Under Richards’s (2007) and Chomsky’s (2007) view, this feature never has a chance to be deleted within the next phase, since the next phase head has no access to the information that this valued feature is in fact an uninterpretable feature that has been valued in the previous phase. In other words, for Richards (2007) and Chomsky (2007), the structure in (22) is destined to crash in the later course of the derivation.

One thing important to notice about configurations (21) and (22), instances with no feature-inheritance, is that, unlike (19), whether they converge or crash is not determined at this phase, since nothing problematic occurs in

---

10 I am not sure whether this derivation counts as a “crashing” one or as a converging but semantically abnormal one. Though crash is defined by unvalued features reaching to interfaces, whether valued (hence indistinguishable from interpretable counterparts) features cause “crash” is unclear. In any case, such a derivation causes a problem in interpretation at the interfaces. I thank an anonymous EL reviewer for letting me clarify this point.
their internal domain (i.e. the object transferred to the interfaces). Rather, their ultimate status should become clear only in the next phase. Because of this property, these two configurations begin to assume a theoretical meaning in light of the discussion on DP-internal syntax given in Section 2.

Recall the problems of DP-internal distribution of features given in (13b, c), and compare them with the configurations above in (21)–(22). One can easily figure out that the configuration in (21) corresponds to the situation involving a [uCase] feature in (13c), and that the configuration in (22) corresponds to the situation involving [uNum] and [uGend] features on D0, as stated in (13b). Thus my conjecture is that neither (21) nor (22) automatically leads to crash of the derivation, as claimed by Richards (2007) and Chomsky (2007, 2008), but rather that they have a potential to converge, and that the DP structure instantiates that possibility. In this view, [uCase] is an uF that is left uninherited and unvalued in the phase in which it first occurs, as depicted in (21). In a similar vein, the “stacked up” [uNum] and [uGend] can be seen as features that are not inherited, but are valued in-situ on PH, as in (22).

If this line of argument is on the right track, we have a syntactic structure for a DP as given in (23).

(23)

In this structure, [uNum] and [uGend], which originate on D0, are not inherited, and they probe down in the internal domain from this position. [uCase], on the other hand, does not probe at all at this point, and serves as the activator of the D0 in later stages of the derivation.11 The present analysis requires modification of the theory of feature-inheritance by Richards (2007) and Chomsky (2007, 2008), with respect to the cases in which uFs are not inherited. The revised conclusions of feature-non-inheritance are given in (24).

11 Or it might be the case that [uCase] also probes down in the DP-internal domain, jointly with [uNum] and [uGend], but fails to find an appropriate goal there. Even if this is the case, we still have the same result with regard to the DP-external operations afterwards: [uCase] counts as an activator of the D0.
(24) **Feature-Non-Inheritance:**
Uninherited features do not cause automatic crash at that phase. Rather,
a. A valued uF causes crash in the next phase only if the identical feature is interpreted elsewhere. \((contra \ (18d))\)
b. An unvalued uF causes crash in the next phase only if it is not valued in that phase. \((contra \ (18c))\)

The condition in (24a) requires special attention, since it apparently leads us to abandon the valuedness-interpretability correspondence of features, which states that all and only interpretable features are valued. If \([u\text{Num}]\) and \([u\text{Gend}]\) on \(D^0\), which are valued by probe-goal Agree at the DP level, survive in the later course of the derivation, why do they not cause interpretive problems at the CI interface? Here we need a precise characterization of the identity of features. I would like to put forth the hypothesis in (25), the **structured \(\phi\)-feature hypothesis**.

(25) **Structured \(\phi\)-feature:**
A \(\phi\)-feature is structured; interpreting the sub-features \([\text{Per}]\), \([\text{Num}]\), \([\text{Gend}]\) separately does not imply interpreting the \(\phi\)-feature as a bundle.

In this analysis, number and gender features are “interpreted” twice. First, interpretable \([\text{Num}]\) and \([\text{Gend}]\) are interpreted separately within a DP; second, valued \([u\text{Num}]\) and \([u\text{Gend}]\) are interpreted as part of a complete \(\phi\)-feature on \(D^0\). In this sense, we are partially relaxing the bijective relation between valuedness and interpretation of formal features (cf. Pesetsky and Torrego (2007)). In a nutshell, the features that drive DP-internal Agree operations are separate, unbundled \([u\text{Num}]\) and \([u\text{Gend}]\) features. Each of the interpretable sub-features \([\text{Per}]\), \([\text{Num}]\) and \([\text{Gend}]\) is interpreted separately on \(D\), \(\text{Num}\), and \(\text{Class/N}\), but this procedure is distinct from the one interpreting the \(\phi\)-feature on the DP as a whole, that is, DP as an argument.\(^{12}\) DP then can be defined as a phase in which an argument is built up, similarly to the way we define a \(v^*P\) phase as a domain for argument-structure licensing, and a CP phase as a domain for information-structure licensing (see Chomsky (2000, 2001)). The \(\phi\)-feature on \(D^0\),

\(^{12}\) Akira Watanabe (personal communication) suggested another way to deal with this valuedness-interpretability split problem, along the line of Harbour’s (2011) model of feature valuation. I leave the task of working out the mechanism for future research, but see Harbour (2011) and Watanabe (2012) for an interesting conceptualization of a feature valuation mechanism.
which has acquired its sub-features in its internal domain, in turn counts as a single argument once it enters into the clausal phase.

(26) **DP as an argument-building phase:**

DP constitutes a phase. The argument status of the DP as a whole is guaranteed by the φ-complete D₀, which has acquired the values of [uNum] and [uGend] through DP-internal Agree relations.

(27) **Phi-feature on DP, split inside, bundled outside:**

Valued [uNum] and [uGend] on D₀ do not cause crash in the clausal (C and v*) phases because there are no identical φ-features interpreted within the DP it heads.

The present analysis immediately ushers in various theoretical questions. An anonymous EL reviewer questions why the multiple interpretation of the “same” features, namely [Num] and [Gend], as claimed here, does not count as a violation of the principle of Full Interpretation, especially in the rigidly phase-based model. First of all, the derivation pursued here does not violate FI. FI, as I conceive of it, requires that all and only features that reach the interfaces get some interpretation. To the extent this way of formulating FI is correct, the features at issue are not themselves illegitimate objects at the interfaces. They are valued (whether inherently or computationally), they reach the interfaces, and they are interpreted there.

Secondly, the same reviewer asks whether these features cause a PIC violation, and if not why not, since the computation seems to keep information from the preceding phase. In a sense, however, we can speculate that the structured φ-feature is a manifestation of the PIC effect, on a feature level. The very first question with which we started the whole investigation is why the sub-features of a φ-set act separately inside a DP, but act as a single unit outside the DP. The answer from the present perspective would be as follows. Within the DP phase, each of these sub-features does count as an independent syntactic object, and the computational system acts on them, stacking them up on a single head (i.e. D₀). Once the DP phase is complete, the computational history inside it is “forgotten,” and all we have in the next phase is a single, atomic φ-feature.¹³

In fact, conditions like (26) and (27) must be stated in one way or another, since otherwise, we would have no single object that counts as an argu-

---

¹³ In set-theoretic terms, it might be the case that, at D₀ (or phase heads in general), the union of features is formed out of the sets of features inside its domain.
ment in the computational system. Thus we obtain a schematic structure of DP-internal and DP-external (i.e. clausal) domains as in (28).

\[
(28) \quad \begin{array}{c}
C \\
\text{[uφ]}
\end{array} \xrightarrow{\text{INHERIT}} T \\
\text{[uφ]} \quad \begin{array}{c}
\text{DP} \\
\text{D} \\
\text{[Per]}
\end{array} \xrightarrow{\text{[uNum]}} \text{[Num]} \xrightarrow{\text{[uGend]}} \text{[uCase]}
\end{array}
\]

Before leaving this section, let us look at what happens if ϕ-features on clausal phase heads (C and v*) are uninherited and valued in-situ, and reconsider why the feature-inheritance corollary (18) seemed to make sense in a clausal domain. Compare the structures in (29) and (30); C-to-T inheritance occurs in the former case, and does not in the latter.

\[
(29) \quad \begin{array}{c}
\text{C-T Feature-Inheritance:} \\
\text{[uφ]}
\end{array} \xrightarrow{\text{INHERIT}} T \\
\text{[uφ]} \quad \begin{array}{c}
\text{DP} \\
\text{D} \\
\text{[Per]}
\end{array} \xrightarrow{\text{[uNum]}} \text{[Num]} \xrightarrow{\text{[uGend]}} \text{[uCase]}
\end{array}
\]

\[
(30) \quad \begin{array}{c}
\text{* Feature-Non-Inheritance:} \\
\text{[uφ]}
\end{array} \xrightarrow{\text{INHERIT}} T \\
\text{[uφ]} \quad \begin{array}{c}
\text{DP} \\
\text{D} \\
\text{[Per]}
\end{array} \xrightarrow{\text{[uNum]}} \text{[Num]} \xrightarrow{\text{[uGend]}} \text{[uCase]}
\end{array}
\]

In (29), the derivation in the CP phase proceeds exactly in the way Chomsky (2007, 2008) and Richards (2007) propose. The [uφ] feature on C is inherited by T and it probes down from this T position, eventually finding D⁰ as an appropriate goal. As part of Transfer at this CP phase, the valued [uφ] on T is successfully deleted.

On the other hand, in (30), the [uφ] feature on C is not inherited, and probes down directly from this position. Probe-goal Agree itself is successful even in this case, since the [uφ] on C finds an appropriate goal on D⁰. The problem shows up when we perform Transfer at this phase: the valued [uφ] on C remains untransferred and survives in the next phase. Unlike the case of the DP phase, in this configuration we have the identical ϕ-feature interpreted elsewhere, namely on D⁰. Thus the com-
putational system fails to keep track of which features to interpret at the CI interface, and the derivation crashes in the next phase. Here conditions (24a) and (25) jointly play a crucial role in attaining the desired result. By the same token, the mode of explanation that is used to exclude (30) demands that the relevant feature on C-T be an atomic, unstructured \( \phi \)-feature. A theoretical implication that can be drawn from the present analysis is that clausal phases (CP and v*P) and nominal phases (DP) behave differently with regard to feature-inheritance, and that the clause-nominal parallelism breaks down at least in this respect.

Another aspect of the clause-nominal asymmetry has to do with the notion of “feature identity” as exploited here. That is, how it is possible for the computational system to distinguish a pair of “identical” \( \phi \)-features that result from Agree, especially when they occur in different phases, which should be banned in the present approach, from different \( \phi \)-features that happen to have the same value, which of course should be allowed. Consider the following simple example.

\[(31) \quad \text{He thinks that he is honest.}\]

The interpretation at stake is the one with the free reading of the lower pronoun (throughout this article I am putting aside the problem of bound pronouns). The problem is that, in cases like (30), on the one hand, the \( \phi \)-feature on C and the \( \phi \)-feature on D should count as “identical” and cause crash; in cases like (31), on the other hand, the \( \phi \)-features on the higher \textit{he} and on the lower \textit{he} should count as distinct, albeit they presumably have the same \( \phi \)-values. How can it be the case, given that these two configurations are parallel in that both involve two \( \phi \)-features occurring in different phases, and given that information in the previous phases is forgotten by the computational system? Here I suspect that the solution would be to incorporate the notion of index/referentiality into the structure of argument DPs. In a nutshell, argument DPs, or at least those that denote real-world entities, carry an index as to which entity they refer to, in addition to the

---

14 Again, the ultimate fate of this derivation is somewhat vague (see note 10). No matter how the problem is to be addressed, at the present stage of our understanding, I believe the problem would recur in any approaches that adopt the phase-based Agree system. What is important for the present purpose is that, in the latter case, the Agree relation is established between the identical (atomic) \( \phi \)-features, whereas in the former, the Agree relation is “partial,” in that [Num] and [Gend] work as independent features.

15 I owe the elaboration of the discussion on feature identity in this paragraph, as well as the example in (31), to an anonymous EL reviewer. I am aware that the solution suggested in the text is tentative, though.
formal φ-features (Inokuma (2011)). The higher he and the lower he in
(31) would then be distinct with regard to this Index feature, whereas the C
in (30) would remain defective in the sense that it lacks an index as charac-
terized here. In other words, the valued φ-feature on C would count as an
“argument without an index,” leading to interpretive problems. Additional
assumptions and further clarifications seem to be in order, but the task
goes far beyond the scope of this article.

4. Concluding Remarks

In this article I have pointed out several theoretical gaps that lie between
clausal and nominal syntax with regard to the treatment of φ-feature inside
and outside DP. It has been shown that in order to stack up the sub-fea-
tures that are scattered within DP and to make them available to the clausal
domain as a single φ-feature, we need a more sophisticated mechanism of
the probe-goal Agree relation. For that purpose, I have tried to add as little
as possible to existing theories. Through a careful examination of the the-
ory of feature-inheritance put forth by Chomsky (2007, 2008) and Richards
(2007), I have claimed that there should exist cases in which uninherited
features do not cause automatic crash, and that the syntax of the DP-internal
domain is the realization of this possibility. In the course of discussion, I
have also argued that interpreting a φ-feature as a bundle and interpreting its
sub-features separately count as different procedures for the computational
system (the structured φ-feature hypothesis). This series of considerations
eventually leads us to the conclusion that DP is a phase, where the argu-
ment status of DP is built up through valuation of the sub-features of the
φ-set.

Seen from a broader theoretical perspective, the discussion in this article
has raised more questions than answers. Let us close our discussion with
a few examples. Our focus has been on the treatment of φ-features inside
and outside DPs, and the two assumptions we have been trying to keep
intact, in the name of theoretical parsimony, are the φ-Completeness Condi-
tion and the Activity Condition. If the structured φ-feature hypothesis is
correct, and DP-internal Agree can be characterized as a stacking-up process
of (sub-)features, then we can (and should) ask the following question: are
there any equivalent processes in other domains, especially in the clausal

16 Questions remain, for instance, as to the treatment of clausal arguments.
domains? An anonymous EL reviewer raises exactly this question, giving as possible “clausal sub-features” Topic, Focus, and Force (cf. Chomsky’s (2008) remarks on left periphery cited in footnote 1). I see no relevant evidence at this point, either theoretical or empirical. However, the underlying idea behind the present investigation is that arguments, syntactically realized as DPs, must be available as single, atomic objects to the domain that computes argument structure, namely, the clausal domain. In other words, in building up its argument structure, the clausal domain does not care about what has happened inside each of its arguments (recall the above speculation on the featural PIC effect). If so, what we have to look into in light of “clausal sub-features” would be the cases where argument structure (built up in v*P phases) and propositions (built up in CP phases) in turn function as “arguments” in higher phases. First candidates are constructions such as embedded clauses, Raising/ECM constructions, and restructuring constructions, the topic of which I have to leave for future research.

An independent but interrelated question crops up about the nature of the Activity Condition, too. To my knowledge, little has been discussed in the literature about whether the Activity Condition is operative in domains other than φ-Agree (and occasionally wh-Agree). If the answer to this question is positive, we would have far more uninterpretable features than we routinely postulate (see also footnote 6). If not, we have to limit and relativize the Activity Condition accordingly.

Pursuing the question about the Activity Condition inevitably leads us to contemplate the nature of Case features, the activator of DP arguments. I have suggested that [uCase] can be seen as a uF that is neither inherited nor valued at the phase it first occurs in, and functions as an activator of the D0 in the next phase. In the present perspective, they are treated in such a manner as in (32).

(32) Case features on D:
Case features on D0 are not inherited, nor are they valued in-situ via downward probing in the DP phase; rather, they must be valued in the next phase.

On the nominal side, Case features are normalized to a certain extent, in

17 More specifically, v*P phases (Chomsky (2000, 2001)).
18 Here, the temporal dimension seems to be sneaking into the generative system (cf. Chomsky (2007: 6)). See Inokuma (2006) for some preliminary remarks.
19 And finally, root clauses.
that now they can be seen as an instantiation of feature-non-inheritance, as predicted by the relativized feature-inheritance theory.\textsuperscript{20} However, a longstanding question of Case remains unsolved. Case is radically uninterpretable; we have not found the interpretable counterparts of \([u\text{Case}]\) features.\textsuperscript{21} Unveiling the nature of structural Case is an enormous mission for any linguistic theory, and I leave it for a future, long-term project.

REFERENCES


Fox, Danny and David Pesetsky (2005) “Cyclic Linearization of Syntactic Structure,”

\textsuperscript{20} Treating \([u\text{Case}]\) this way predicts that there are no DP-internal Case-agreement phenomena. This prediction is simply false, as shown by the well-known Case-concord phenomena found in languages like Greek. As it stands, the present theory has nothing to say about the solution and I will leave the problem for future research.

\textsuperscript{21} Recent efforts to reduce Case to some interpretive primitives can be found in Pesetsky and Torrego (2004), Manzini and Savoia (2011), and Sigurðsson (2012).
Theoretical Linguistics 31, 1–45.
Department of English
Jissen Women’s University
4–1–1 Osakaue, Hino-shi
Tokyo 191–8510
e-mail: sakinokuma@gmail.com