

Doreen Georgi

A Local Derivation of Global Case Splits*

Abstract

In this paper I argue that Global Case Splits can be derived in a local and cyclic way within a minimalist framework. I show that the problems which the phenomenon poses for strictly derivational models of syntax (look-ahead, counter-cyclic operations) can be solved by means of the operation *Maraudage*. Furthermore, I show how the analysis of Global Case Splits can also account for Local Case Splits, languages without case splits and for direction marking.

1. Introduction

The textbook examples of non-local dependencies are usually restricted to the domains of movement, agreement, and binding. The important syntactic domain of case assignment, however, is usually assumed to be local, apart from ECM constructions. From the viewpoint of a derivational syntax, however, ECM does not pose a severe problem because the case assigner in the matrix clause c-commands the target of case assignment in a sufficiently local domain. The aim of this paper is to show that there are indeed non-local dependencies for case assignment which do not seem to be compatible with a derivational bottom-up model of syntax that does not allow for look-ahead and counter-cyclic operations: Global Case Splits. Although the data have already been introduced by Silverstein (1976), this phenomenon has barely been discussed in the theoretical literature on case assignment, even less from a syntactic locality perspective. I will show that it is possible to derive these global splits in a minimalist derivational framework. In section 1 I sum up the characteristics of Global Case Splits. I continue by illustrating why they are a challenge for a strictly local derivational syntactic theory. In section 3 I outline my analysis which is based on a shift of perspective on the data. Afterwards, I go through the derivations in

* For valuable comments I would like to thank Anke Assmann, Sebastian Bank, Stefan Keine, Antje Lahne, Marc Richards, Martin Salzmann, Jochen Trommer, and Eva Zimmermann as well as the audiences at ConSOLE XVII (Nova Gorica, January 2009), the ‘Repairs’ workshop at the DGfS (Osnabrück, March 2009), the Potsdam-Leipzig-Workshop (Wittenberg, March 2009), and MASL (Budapest, August 2009) for their remarks and comments on earlier versions of this paper. I am indebted to Fabian Heck and Gereon Müller for extensive discussion of the topic. Georgi (2009) contains an earlier version of parts of this work.

some languages with Global Case Splits and I show how variation between these languages can be accounted for. Finally, in section 5 I extend the analysis to languages with Local Case Splits, languages without case splits and to direction marking.

In a non-split case system an argument A in syntactic position X always bears case K, independently of the inherent features of A or the presence and features of its coargument(s). In split case systems, however, the case of A depends on the properties of A and/or its coargument(s). The choice is driven by Silverstein scales (Hale (1972); Silverstein (1976)), cf. (1): Arguments high on a scale are encoded differently from those lower on the same scale, a phenomenon called *differential argument encoding*.

(1) *Silverstein scales (Silverstein (1976))*

- a. person scale: 1st > 2nd > 3rd
- b. animacy scale: human > animate > inanimate
- c. definiteness scale: pronoun > proper name > definite > indefinite specific > non-specific

Silverstein (1976) distinguishes between local and global splits, following the terminology of Chomsky (1965). In languages with *Local Case Splits* (LCS), the case marking of an argument of a transitive verb solely depends on the properties of the internal argument of the verb. In Hebrew, for example, the internal argument of a transitive verb is marked by the case prefix *?et* if it is a pronoun, a name or definite, i.e., high on the definiteness scale. In any other environment, it is zero-marked:

(2) *Local Case Split in Hebrew (Aissen (2003, 448))*

- a. Ha-seret her?a ?et-ha-milxama
DEF-movie showed ACC-DEF-war
'The movie showed the war.'
- b. Ha-seret her?a (*?et)-milxama
DEF-movie showed (ACC-)war
'The movie showed a war.'

Global Case Splits (GCS) differ from LCS in that case marking of an argument of a transitive verb does not only depend on its own properties but also on those of its coargument. In the languages analysed in this article, both arguments of a transitive verb are usually zero marked, but if the internal argument (DP_{int}) is higher on a Silverstein scale than the external argument (DP_{ext}), one of both arguments bears an overt case marker. In Yurok, for example, DP_{int} bears an overt case marker if it is higher on the binary person scale in (3) than DP_{ext} .

(3) *Person hierarchy in Yurok*

- 1st/2nd > 3rd person.

(4) *GCS in Yurok (Robins (1958, 21))*

- a. keʔl nek ki newoh-paʔ
 2SG.NOM 1SG.NOM FUT see-2>1SG
 ‘You will see me.’ 2nd > 1st
- b. yoʔ nek-ac ki newoh-peʔn
 3SG.NOM 1SG-ACC FUT see-3SG>1SG
 ‘He will see me.’ 3rd > 1st

Languages with GCS vary in two respects:

- (i) the relevant scale and
 (ii) the realization of the case split on either DP_{ext} or DP_{int} .

The following table shows a survey of GCS languages:

Table 1. *Languages with Global Case Splits*

Language	Reference
Arizona Tewa (Kiowa-Tanoan)	Kroskirty (1978; 1985)
Awtuw (Sepik-Ramu)	Feldman (1986)
Fore (Trans-New Guinea)	Scott (1978)
Kashmiri (Indo-European)	Wali & Koul (1997)
Kolyma Yukaghir (Yukaghir)	Maslova (2003)
Umatilla Sahaptin (Penutian)	Rigsby & Rude (1996)
Yurok (Algic)	Robins (1958)

There is a large body of literature on Local Case Splits (cf. among others Silverstein (1976); Comrie (1979); Lazard (1984); Bosson (1985); Aissen (1999; 2003); Keine & Müller (2010)), but there are only very few formal approaches to Global Case Splits. This is remarkable given that the latter are more problematic for derivational syntactic theories: it seems that the decision which case to assign needs a non-local representation of structure that includes both coarguments and the case assigner, hence the name ‘global’ split.

2. Global Case Splits and locality: a challenge

Global Case Splits like the one in Yurok are called ‘global’ because it seems that the case assigner must be able to have access to the properties of two arguments in order to be able to decide which case to assign to one of them. Recent minimalist syntactic approaches, however, try to reduce globality and to model restrictions within small subparts of the derivation. Therefore, GCS impose serious problems for a derivational syntactic theory like minimalism. Before I can illustrate this point, I briefly sum up standard minimalist assumptions about case

assignment in transitive contexts and structure building in general (Chomsky (1995; 2000; 2001)).

(5) *Structure building and case assignment in minimalism*

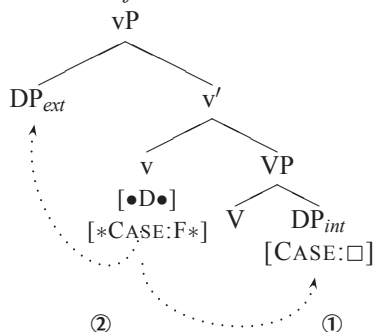
- a. Syntactic structure unfolds step by step in a bottom-up fashion.
- b. All operations are in accordance with the *Strict Cycle Condition* (cf. (6)).
- c. Structure-building (Merge) is feature-driven (by c-selection features represented as [$\bullet F \bullet$]).
- d. DPs enter the derivation with an unvalued case feature [$\text{CASE}:\square$] that has to be valued by Agree with a c-commanding functional head.
- e. v has a dual role: it assigns case to DP_{int} and selects DP_{ext} .
- f. T assigns case to DP_{ext} .

(6) *Strict Cycle Condition (SCC, based on Chomsky (1973))*

- a. No operation can apply to a domain dominated by a cyclic node α in such a way as to affect solely a proper subdomain of α dominated by a node β which is also a cyclic node.
- b. Every projection is a cyclic node.

(7) *Structure of transitive vP*

- ①: v assigns Case value F to DP_{int}
- ②: v selects DP_{ext}



This system has been developed on the basis of languages without case splits like English where an argument is assigned a case value independently of its coargument. However, if we try to derive a Global Case Split in the same way, a dilemma arises: There are two possible derivations depending on which operation-inducing feature on v is discharged first (the case assigning feature or the c-selection feature), but each derivation violates some core principle of a strictly derivational framework. Let me illustrate this on the basis of a global split like the one in Yurok on DP_{int} .

- (i) ① \succ ② (case assignment precedes c-selection): v assigns case to DP_{int} directly after it has merged with VP. But the case value of DP_{int} (Nom vs. Acc) also depends on the properties of DP_{ext} which has not yet been

merged. Hence, case valuation would need **look-ahead**, which is impossible in a strictly derivational syntax.

- (ii) ② > ① (c-selection precedes case assignment): In order to circumvent the look-ahead problem one could assume that the order of operations induced by v is reversed such that DP_{ext} is merged *before* v values case on DP_{int} . But then case valuation is **counter-cyclic** given the strictest version of the *Strict Cycle Condition* in (6): case assignment affects only v' although this projection is already dominated by vP .

Thus, no matter which order of operations is chosen, none is in accordance with minimalist assumptions about locality and cyclicity.¹

Apart from the look-ahead and the cyclicity problem further issues arise: Somehow v must be able to compare the properties of DP_{int} and DP_{ext} in order to decide which case to assign. The question is thus how v communicates with two arguments. Finally, there must be a mechanism which fixes the case value that v assigns (Nom or Acc in Yurok). There could for example be a feature-changing operation in the syntax (see Noyer (1998)) or the case feature is inserted after v has compared the properties of the coarguments.

Previous analyses of GCS include Aissen (1999); De Hoop & Malchukov (2008); Keine (2010). But each of them faces at least one of the problems mentioned above. The most problematic component of these approaches is that they are all global in the sense that the decision which case to assign is made on the basis of a representation which includes both arguments of a transitive verb and rules/constraints which make reference to both of these arguments. As a result, it is necessary in some cases to apply case assignment counter-cyclically. Béjar

¹ A different problem comes up for languages with the case split on DP_{ext} . There is no look-ahead or a counter-cyclic operation because when DP_{ext} receives its case value, both arguments are merged and hence potentially accessible for the case assigner of DP_{ext} , the functional head T (both are in the c-command domain of T). However, if the strict version of the *Phase Impenetrability Condition*, which restricts the search space of a probe, is adopted, it is impossible for T to access DP_{int} :

- (i) *Phase Impenetrability Condition (PIC, Chomsky (2001))*
- a. In a phase α with the head H, the domain of H is not accessible to operations outside α , only H and its edge are accessible to such operations.
 - b. The domain is the complement of a phase head, the edge is its specifier.

Under standard assumptions v is a phase head. This means that case assignment from T to DP_{ext} cannot refer to the properties of DP_{int} because DP_{int} is in the domain of v and hence no longer accessible as soon as vP is completed. There might be several solutions for this problem that are independently motivated, e.g. movement of DP_{int} to the phase edge or cyclic Agree via v (Legate (2008)). Hence, with respect to locality, splits on DP_{int} are more problematic for a derivational account than splits on DP_{ext} .

& Řezáč (2009) also develop a local account but it is not clear to me how the case value that *v* assigns is fixed.

3. Analysis

3.1. A new perspective

The desideratum is to derive GCS without violating the SCC in (6), but this means that we face the look-ahead problem. In order to circumvent it, I propose that the data should be looked at from a new perspective. GCS has always been described in a way that the case value of an argument depends on the properties of two coarguments and this is what brings about the global character of the phenomenon. I suggest that the data can also be characterized as follows:

(8) *A different perspective on GCS*

It is not case marking that depends on the properties of the coarguments. Rather, the properties of DP_{int} determine what properties DP_{ext} can have. This means that the selection properties of *v* are restricted by the properties of DP_{int} .

Consider the case of Yurok in (4), where DP_{int} bears an overt case marker (called ‘accusative’) if it is higher on the person hierarchy in (3) than DP_{ext} . All combinations are displayed in the following table:

Table 2. GCS in Yurok

person of DP_{ext}	case of DP_{ext}	person of DP_{int}	case of DP_{int}
1st/2nd	Nom	1st/2nd	Nom
3rd	Nom	1st/2nd	Acc
1st/2nd	Nom	3rd	Nom
3rd	Nom	3rd	Nom

Under the new perspective GCS in Yurok can be described as follows: If DP_{int} is 1st/2nd person nominative, DP_{ext} has to be 1st/2nd person as well; if DP_{int} is 1st/2nd person accusative, DP_{ext} has to be 3rd person; if DP_{int} is 3rd person, there are no restrictions on the person of DP_{ext} . This means that the person of DP_{ext} is the dependent feature, not the case of DP_{int} . A local analysis is now possible if there is a way to let the first merged argument DP_{int} influence the properties of DP_{ext} , depending on the case and person features of DP_{int} .

Two questions arise in this context: (i) How can DP_{int} restrict the nature of DP_{ext} ? (ii) What drives the occurrence of the overt case marker? My answer to the first question is that there is a repair operation, called *Maraudage*, which

steals features originally provided for DP_{ext} depending on the features of DP_{int} . With respect to the second question I propose that the case marker is a reflex of Maraudage which is realized postsyntactically.

3.2. Assumptions

In this subsection I summarize my theoretical assumptions which are the basis for the derivations in the next section.

I assume a strictly derivational model of syntax with the following properties (cf. (5)): The syntactic derivation unfolds bottom-up in accordance with the *Strict Cycle Condition* by alternating applications of the basic operations Merge and Agree. All operations are feature-driven ($[\bullet F \bullet]$ triggers Merge, $[*F*]$ triggers Agree; for the notation cf. Sternefeld (2006); Heck & Müller (2007)). v agrees with DP_{ext} and DP_{int} in phi-features (cf. the *two arguments against one head*-configuration in Anagnostopoulou (2003); Adger & Harbour (2007); Heck & Richards (2010); Řezáč (2008); Richards (2008); Béjar & Řezáč (2009); Keine (2010)). In order to be able to agree with two arguments v provides two sets of probe features $[*F*]$: one for checking with DP_{ext} and another one for checking with DP_{int} . A v in a transitive context thus has the following features when it enters the derivation:

$$(9) \{ [\bullet V \bullet] \succ [\bullet D \bullet], [*F*]_{ext}, [*F*]_{int} \}$$

First, v wants to merge with VP and it selects a DP, the external argument. Furthermore, it provides a probe for Agree with DP_{ext} ($= [*F*]_{ext}$) and for Agree with DP_{int} ($= [*F*]_{int}$).

The operation Agree is defined as follows (based on Chomsky (2000; 2001)):

(10) *Agree*

Agree between a probe P and a goal G applies if

- a. P c-commands G.²
- b. G is the closest goal to P.
- c. P and G have matching feature values (Match = feature identity).
- d. P and G have matching set indices.
- e. Result: P and G check their matching features.

This is the standard definition of Agree in which a probe with operation-inducing features is checked by the closest matching goal in its c-command domain. What

² In the present analysis v must enter into an Agree relation with DP_{ext} . I assume percolation of features from v to v' so that v can c-command DP_{ext} . Alternatively, one could replace 'c-command' in the definition of Agree with 'm-command'. Nothing in the analysis depends on the choice between the two options.

is added for the purposes of this article is condition (10-d). Given the assumption that v has two probe sets, one for each argument, an argument can only check the features of the set which is coindexed with it, i.e., DP_{ext} can only check features in probe set $[*F*]_{ext}$ and DP_{int} can only check features in probe set $[*F*]_{int}$.³ Note that in this definition Match is a prerequisite for Agree and therefore has to apply *before* the actual Agree operation (checking) takes place. This will become relevant in what follows.

Merge and Agree are triggered by the need to check operation-inducing features (c-selection features and probe features) as demanded by the principle *Full Interpretation* (Chomsky (1995)). If a clause contained unchecked features at the end of a derivation, it would not be interpretable at the interfaces. I assume that not only operation-inducing features must be checked, but in addition, also certain phi-features of the arguments must enter into an Agree relation in order to get checked. Which phi-features are subject to this constraint depends on the Silverstein scale which drives the split. In Yurok, for example, the person scale is relevant for the case split and hence, person features of the DPs must enter into Agree with v . This requirement is formulated in the constraint FEATURE CHECKING (cf. the *Person Licensing Condition* in Béjar & Āezáč (2009)):

(11) FEATURE CHECKING (FC)

Goal features have to be checked (person, animacy, ... depending on the relevant scale in a language).

As a consequence, *Full Interpretation* does not only hold of operation-inducing features but also of goal features:

(12) FULL INTERPRETATION (FULLINT)

A clause must not contain unchecked features (c-selection features, probe features, goal phi-features).

Furthermore, I follow Béjar (2003); Béjar & Āezáč (2009), based on Harley & Ritter (2002), in that inherent features of DPs like 1st person, [+animate], etc. are complex objects which are decomposed into privative features and represented by bundles of these privative features (for the same basic idea but with different privative features cf. Harbour (2008)). They argue for a decomposition of person

³ The indices of the probe sets are sufficient for the purposes of this article, but they might be problematic when Agree / case assignment over clause boundaries is considered, e.g. when v Agrees with an argument of an embedded clause in ECM constructions. This argument may be the external argument DP_{ext} of the embedded clause and then the internal probe set could not match with DP_{ext} due to different set indices. A solution which can be adopted for case splits and ECM would be to order the probe features on a stack such that the set with fewer features is the highest probe set and the probe set with more features is below this set. Since only the highest feature on a stack is accessible for operations, the lower set can only trigger operations if the first set is checked and deleted. In this way no indices are necessary to account for the order in which the probe sets are discharged and ECM is not problematic anymore.

into three, in part semantically motivated, privative features. There is a general person feature π which differentiates person from e.g. number or animacy. The feature [Participant] encodes speech act participants (1st and 2nd person) and [Speaker] encodes the speaker of a speech act (1st person). These features are abbreviated as [1], [2] and [π], respectively.

$$(13) \quad \underbrace{\underbrace{\underbrace{1st\ person \quad 2nd\ person \quad 3rd\ person}_{[speaker] \cong [1]}}_{[participant] \cong [2]}}_{[person] \cong [\pi]}$$

The traditional person values 1st, 2nd, and 3rd person are then represented as bundles of these privative features:

$$(14) \quad 1st: \begin{bmatrix} [\pi] \\ [2] \\ [1] \end{bmatrix} \quad 2nd: \begin{bmatrix} [\pi] \\ [2] \end{bmatrix} \quad 3rd: \begin{bmatrix} [\pi] \end{bmatrix}$$

The important point of this decomposition is that there are entailment relations between the privative features: If a category contains [Speaker], it also contains [Participant] and [π]. In this way, hierarchies are encoded in the representation of phi-features: A value which is high on a scale is encoded by a superset of features compared to a value which is lower on this scale, in the case at hand this means 1st \succ 2nd \succ 3rd person. I adopt this decomposition and will apply its logic also to other goal features. However, I will represent the privative features by more abstract letters (instead of numbers or abbreviations of their semantics) in order to allow for comparison of patterns between languages in which different phi-features are responsible for the case split (cf. section 4.3). For the person features of Béjar & Řezáč (2009) this looks as in (15).

- (15) *Abstract person features*
- a. [C] = general person feature ($\cong \pi$)
 - b. [B] = participant feature (\cong [Participant])
 - c. [A] = speaker (\cong [Speaker])
 - d. [C] = 3rd person, [BC] = 2nd person, [ABC] = 1st person

Finally, I assume that v has expectations about the properties of its arguments: It expects the typical unmarked case that DP_{int} is lower on the hierarchy than DP_{ext} . In the present system, this means that the probe features which v provides in the internal probe feature set are a subset of the probe features in the external probe feature set. In Yurok, where a distinction is made between local person and non-local person, v enters the derivation with the following features (local person encoded by [BC], non-local person by and [C]; the feature [A] is irrelevant because 1st and 2nd person behave alike with respect to the case split):

- (16) *v* in *Yurok*
 $v \{ [\bullet V \bullet] \succ [\bullet D \bullet], [*BC*]_{ext}, [*C*]_{int} \}$

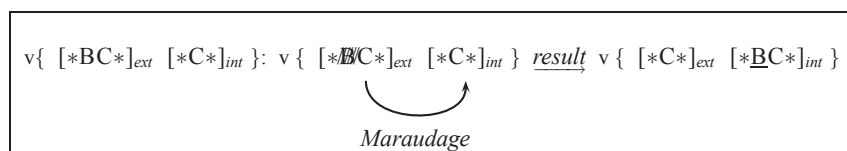
These assumptions have the following consequences: Because of incremental structure building, *v* agrees first with DP_{int} at a stage of the derivation where DP_{ext} has not yet been merged. If DP_{int} is atypical in that it possesses more features than *v* provides for it (viz., if DP_{int} is higher on a scale than expected), it cannot check all of its features and violates FC. Take *Yurok* as an example; *v* is repeated in (17).

- (17) *v* in *Yurok*
 $v \{ [*BC*]_{ext}, [*C*]_{int} \}$

v expects DP_{int} to be 3rd person [C], but if it is 1st or 2nd person [BC], the feature [B] of the goal can not be checked and the constraint FEATURE CHECKING (FC) will be violated. I propose that there is a repair strategy, called *Maraudage*, which can apply in order to avoid the violation of FC: *v* possesses the required probe feature [*B*], but it is in the wrong probe set, the set provided for Agree with DP_{ext} . What happens is that the required feature is displaced into the probe set for Agree with DP_{int} .

- (18) *Maraudage*: Features on *v* can be displaced from probe set_A to probe set_B.

This means that features which were originally provided for checking with DP_{ext} are displaced from set $[*F*]_{ext}$ into set $[*F*]_{int}$. Afterwards, DP_{int} can check the displaced feature in $[*F*]_{int}$ as well:



If *Maraudage* is to lead to the satisfaction of FC for DP_{int} , a certain order of operations is to be adhered to: First, every privative probe feature that wants to enter into an Agree relation looks for a matching goal, because Match is a prerequisite for Agree. If it finds a goal but this goal has a superset of features of the probe and hence there is no one-to-one relation between probe features and matching goal features, *Maraudage* can apply. Afterwards, the actual Agree operation takes place which checks the involved features. In this way it is guaranteed that the marauded feature can enter into an Agree relation with DP_{int} , too, which is the desired result. Note that *Maraudage* cannot apply freely; it is a repair strategy that only takes place when it is necessary to satisfy FC, but it is usually prohibited. This is expressed in the following constraint:

(19) NOMARAUDAGE (NOM)

Do not displace probe features from probe set_A to probe set_B.

The fact that Maraudage is a repair operation suggests an optimality-theoretic analysis (Prince & Smolensky (1993)). If NOM is ranked below FC, it is possible to violate NOM and to displace features in order to fulfill FC for DP_{int}.

Whether Maraudage takes place or not has different consequences for what remains in set [*F*]_{ext}. After Maraudage has taken place, only [*B*] remains in set [*F*]_{ext} and therefore, DP_{ext} cannot be 1st or 2nd person [BC]; if it were [BC], the privative feature [B] of DP_{ext} could not be checked because the probe feature [*B*] had been checked by DP_{int} after Maraudage. Hence, FC is violated. If Maraudage does not take place, i.e., if DP_{int} is 3rd person [C], [*BC*] remains in the external probe set and DP_{ext} can be 1st or 2nd person [BC]. In this way, the restrictions on DP_{ext} are brought about by the properties of DP_{int}.⁴

Crucially, FC and NOM are checked at each derivational step in order to guarantee a *local* derivation of GCS (for the motivation of this concept of *extremely local optimization* cf. Heck & Müller (2007)). In particular, because of incremental structure building, there is the stage v' of the derivation to which these constraints apply. At this stage, DP_{int} is the only argument in the structure. It can trigger Maraudage before DP_{ext} is merged. DP_{ext} has to cope with the remaining features. Hence, DP_{ext} depends on the properties of DP_{int}. The output of this first optimization is then the input for the next evaluation, hence, DP_{ext} is merged with the optimal v'-derivation and vP is projected. The constraints FC and NOM apply again, this time at the vP-level, but they can not access DP_{int} which has been part of the previous optimization. Hence, there is no stage in the derivation at which the constraints can evaluate both arguments at the same time. The approach is thus local and not global. In contrast to FC and NOM, the constraint FULLINT does not apply at every stage of the derivation, but only at the phase level, viz. at vP. It cannot apply at v' because v has probe features for DP_{int} and DP_{ext} and the latter can only be checked after DP_{ext} has been merged.

⁴ Cf. also Adger & Harbour (2007) for the idea that an atypical DP_{int} can absorb features on a functional head which were originally needed to select DP_{ext}. For Adger & Harbour such a derivation is bound to crash. In this way, they derive the strong version of the Person Case Constraint (PCC). In the present analysis, however, it depends on the exact properties of DP_{ext} whether the derivation converges or not. A related difference between the two approaches is that the checking of the feature provided for DP_{ext} by DP_{int} is unavoidable in Adger & Harbour's approach but optional in mine. This is due to the fact that the strong version of the PCC that they discuss is not a global phenomenon but rather a local one: whenever DP_{int} (i.e., the accusative marked direct object of a transitive verb) is 1st/2nd person, the PCC arises regardless of the features of the coargument (the dative). The global version of the PCC is the weak PCC in which the emergence of a PCC effect also depends on the properties of the coargument. When I discuss Local Case Splits, Maraudage will be also obligatory, cf. section 5.1. Hence, Adger & Harbour develop a similar idea but they only apply it to local phenomena.

3.3. Morphological realization

In this subsection I address the question of what the overt case marker realizes. The overt case marker shows up when DP_{int} is higher on a scale than DP_{ext} . It is exactly in these contexts that Maraudage can apply. Therefore, I propose that the overt case marker is a morphological reflex of Maraudage.⁵ When Maraudage takes place, the diacritic ‘_’ is generated in a probe set on v , represented as follows if the probe set contains a feature [F]: [F]. Let us assume for the moment that it attaches to the marauded feature, as in the shaded box below (18). This diacritic is passed on (copied) to the argument that checks the displaced feature via Agree. I propose that the overt case marker is the morphologically realization of this diacritic on an argument. This can be modeled in a postsyntactic, realizational model of morphology like Distributed Morphology (*DM*, Halle & Marantz (1993), Halle & Marantz (1994), Harley & Noyer (1999)). In *DM*, syntax operates solely on morphosyntactic feature bundles. Phonological information is added after the syntactic computation. Vocabulary items (VIs) which pair phonological information with morphosyntactic features are inserted into terminal nodes of the syntactic structure in accordance with the *Subset Principle* and *Specificity*: The most specific VI which realizes a subset of the features of the terminal node is inserted. In GCS languages, there is a VI which is sensitive to the diacritic generated by Maraudage, the element which is the overt case marker:

(20) *Vocabulary items*

- a. /X/ ↔ []
- b. Ø ↔ []

When a DP possesses the diacritic (DP_{int} in the Yurok example), the first vocabulary item is inserted because it is more specific than the second. Otherwise, the

⁵ Harbour (2008) develops a similar idea, namely that the head which initiates Agree (V with DP_{int} and v with DP_{ext}) has different expectations on the properties of the arguments: DP_{ext} should have certain phi-features which encode semantic properties, whereas the expectations for DP_{int} are underspecified, viz. V probably selects only for the category N but not for certain phi-features. The idea is then that copying of phi-features takes place if the specifications of the DP and the expectation about the specifications on the head do not exactly match: a highly specified, atypical DP_{int} copies phi-features on V because V has fewer selection features, whereas a modestly specified DP_{ext} receives phi-features provided by v for it. The core point is that an overt case marker is a realization of the copied phi-features. Our analyses are very similar in the sense that the case marker is a realization of displaced phi-features appearing with atypical arguments. In my approach the displacement is on the functional head which Agrees with the arguments whereas in Harbour’s account it takes place between a head and an argument. The crucial difference is that Harbour’s approach derives only local splits; he tentatively extends his analysis to the strong version of the Person Case Constraint which he takes to be a global phenomenon, but the strong version is indeed a local phenomenon, too, it is only the weak version of the PCC which is global (cf. fn. 4).

zero exponent is inserted; alternatively, there is no zero exponent and hence, no vocabulary item can be inserted if Maraudage has not taken place.⁶

As noted in the introduction, GCS languages differ with respect to the location of the split on either DP_{ext} (ergative) or DP_{int} (accusative). I propose that this difference arises as a consequence of where the diacritic is generated: If it is generated in the probe feature set in which the displaced feature ends up (= $[*F*]_{int}$), it is passed on to DP_{int} via Agree and hence, an accusative case marking pattern arises, as e.g. in Yurok. If the diacritic is generated in the set from which the feature is displaced (= $[*F*]_{ext}$), it is passed on to DP_{ext} via Agree and an ergative pattern arises. Hence, either a language marks that the set $[*F*]_{int}$ is atypical in that there are more features after Maraudage than there were originally, or the language marks that something unusual happened to the set $[*F*]_{ext}$ in that features have been displaced from it.

3.4. Intermediate summary

Let me briefly summarize how the problems for an analysis of GCS in a derivational framework laid down in section 2 are solved in the present system. The case assigner v can communicate with two arguments because it provides Agree-triggering probe features for both DP_{ext} and DP_{int} . Look-ahead is no longer needed because DP_{ext} depends on the properties of DP_{int} under the new perspective in (8): DP_{int} determines which properties DP_{ext} can have and this can be modeled in a derivational bottom-up syntax without look-ahead. Finally, case assignment is in accordance with the *Strict Cycle Condition* because the diacritic which is spelled out by the overt case marker is assigned cyclically by Agree. It is generated on v as a consequence of Maraudage when DP_{ext} has not yet been merged. The system as it is presented up to now overgenerates, because there is not always an overt case marker when DP_{int} is high on a scale (see Table 2). The influence of DP_{ext} is illustrated in the next section.

⁶ Note that there is no abstract case in the system developed in section 3.2: Arguments are not assigned a case value in the syntax which can then be morphologically realized or not. Rather, there is just a morphological reflex of the operation Maraudage which can be called morphological case. But in a sense, there is an equivalent of the case filter in my analysis: the standard minimalist implementation of the case filter is that DPs have an uninterpretable case feature [UCASE] which must be checked as a consequence of an Agree relation with a functional head. The requirement that DPs must Agree with functional heads is formulated in the constraint FEATURE CHECKING in (11) in the Maraudage approach. Hence, there is also a licensing condition of DPs, namely that they must check their phi-features with v , the only difference is that this does not additionally result in checking or assignment of a case value.

4. Derivations

Languages with GCS can be divided in (at least) two groups: those which depend on a binary Silverstein scale and those which depend on a tripartite scale. I call the appearance of an overt case marker a binary scale effect in the former and a tripartite scale effect in the latter.

4.1. Binary scale effects

4.1.1. Yurok

In this subsection I go through the derivation of binary scale effects in detail. The first example is Yurok. We have already seen that DP_{int} bears an overt case marker if it is higher on the person scale in (21) than DP_{ext} .

- (21) *Person hierarchy in Yurok*
1st/2nd > 3rd

Table 3. *Person/case combinations in Yurok*

	person of DP_{ext}	case of DP_{ext}	person of DP_{int}	case of DP_{int}
pattern 1	1st/2nd	Nom	1st/2nd	Nom
pattern 2	3rd	Nom	1st/2nd	Acc
pattern 3	1st/2nd	Nom	3rd	Nom
pattern 4	3rd	Nom	3rd	Nom

The split is driven by a binary scale that only distinguishes speech act participants from non-participants, hence, only the general person feature [C] and the participant feature [B] play a role for GCS.

- (22) *Encoding of person in Yurok*
a. 3rd person: [C]
b. 1st/2nd person: [BC]

v expects DP_{ext} to be higher on the person scale than DP_{int} .

- (23) *v in Yurok*
v [[*BC*]_{ext}, [*C*]_{int}]

In order to derive a Global Case Split, Maraudage must be *optional* if it can apply. The reason is that we do not always find an overt case marker when DP_{int} is atypical, i.e., 1st or 2nd person [BC]. Whether it shows up or not depends on the features of DP_{ext} (cf. Table 3), but since the present approach is local and

cannot access the features of DP_{ext} at the stage v' when it is decided whether Maraudage applies, Maraudage must be optional. It is only at the next cycle, vP , that a decision is made whether the candidate that has applied / has not applied Maraudage at v' wins, depending on the features of DP_{ext} . In OT, optionality can be expressed by a tie between the relevant constraints, namely FC, which eventually triggers Maraudage, and NOM, which prohibits Maraudage: $FC \circ NOM$. This is a conjunctive local tie (cf. Müller (2000, ch. 5)) which means that the two constraints form a complex constraint that is violated if one of its subconstraints is violated.

At the vP -level, FULLINT will become decisive. If v or DP_{ext} has an unchecked feature, the derivation must crash in order to derive that some logically possible patterns are not attested. The crash of the derivation can be represented in OT in the following way: there is a candidate which is empty \emptyset , the empty output. If this candidate becomes optimal, nothing is pronounced, which is in a sense the same as the crash of the derivation – a certain combination of features cannot be uttered. The EMPTY OUTPUT CONDITION militates against the empty output:

- (24) EMPTY OUTPUT CONDITION (EOC)
Avoid the empty output.

The final ranking of constraints for languages with a GCS is shown in (25):

- (25) *Ranking in GCS languages*
 $FULLINT_{vP} \gg FC \circ NOM \circ EOC$

I go now through the derivations to show how the patterns in Table 3 are derived in a local way. The output(s) of the optimization at the v' -stage is (are) the input(s) for the optimization at the vP -stage of the derivation. Since $FULLINT_{vP}$ does not apply at v' , I leave it out in the tableaux which evaluate this stage of the derivation.

Let us begin with a typical DP_{int} which is 3rd person [C]. Checked features are indicated by a strike-through [~~±~~]; a marauded feature is represented as [~~±~~] in its original set. ‘_’ is the representation of the diacritic generated by Maraudage.

SCENARIO 1: DP_{int} is 3rd person [C]

(26) *Stage of the derivation* = v'

Input: $v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int}=[C]$	FC	NOM	EOC
☞ $C_1: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int} [C]$			
$C_2: v \{ [BC*]_{ext}, [BC*]_{int} \}$ $DP_{int} [C]$		*!	
$C_3: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int} [C]$	*!		
$C_4: \emptyset$			*!

C_1 in which the only probe feature of v in $[*F*]_{int}$ is checked with the only person feature of DP_{int} is the optimal candidate because it does not violate any constraint. All other candidates violate the constraint tie once: C_4 is the empty output and violates EOC, in C_3 no Agree applies and hence the person feature [C] of DP_{int} is not checked, which violates FC; C_2 marauds a feature but since Maraudage is not necessary in this case, it is blocked by a violation of NOM. C_1 is thus the input for optimization at the vP -level. Since C_1 is a candidate without Maraudage, there will be no overt case marker if DP_{int} is 3rd person. The first case to consider at the vP -level is one in which a 1st/2nd person DP_{ext} is merged with the output of the previous optimization:

(27) *Stage of the derivation* = vP , DP_{ext} is 1st/2nd person [BC]

Input: $v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{ext}=[BC]$	FULLINT	FC	NOM	EOC
☞ $C_1: v \{ [BC*]_{ext}, [*C*]_{int} \}$ $DP_{ext} [BC]$				
$C_2: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{ext} [BC]$	*!*	*		
$C_3: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{ext} [BC]$	*!***	**		
$C_4: \emptyset$				*!

If DP_{ext} is 1st/2nd person [BC], all features of DP_{ext} and v can be checked since v provides exactly the probe feature counterparts $[*BC*]$ and hence, no constraint is violated. If only one of the features or no feature has been checked (C_2 and C_3), FULLINT is fatally violated by the unchecked probe features of v and the unchecked person features of DP_{ext} . The empty output violates EOC.

Next, assume that a 3rd person DP_{ext} [B] is merged with the optimal output C_1 of the optimization in (26). This derivation should crash since the probe feature

[*B*] of v cannot be checked, given that DP_{ext} is [C]. However, this pattern is attested. The crucial observation is that when both arguments of a transitive verb are 3rd person [C], it is already clear before the derivation starts that [*B*] can never be checked because neither DP_{int} nor DP_{ext} possesses a feature [B]; both are 3rd person [C]. I propose that the system is able to detect such a situation and provides a mechanism that solves the problem already in the numeration before the derivation starts:

(28) *F-deletion*⁷

A probe feature [*F*] can be deleted on a head α in the numeration if it is impossible to check F in the first place, because none of the arguments of α possesses a matching feature F (where F is a variable over the privative features A, B, and C).

A feature which is deleted in the numeration is set in gray in the tableaux. The derivation with a 3rd person DP_{ext} is then as follows:

(29) *Stage of the derivation = vP , DP_{ext} is 3rd person [C], F-Deletion applies to [*B*]*

Input: $v \{ [*BC*]_{ext}, [*E*]_{int} \}$ $DP_{ext}=[C]$	FULLINT	FC	NOM	EOC
\Rightarrow $C_1: v \{ [*BE*]_{ext}, [*E*]_{int} \}$ $DP_{ext} [E]$				
$C_3: v \{ [*BC*]_{ext}, [*E*]_{int} \}$ $DP_{ext} [C]$	*!	*		
$C_3: \emptyset$				*!

As in the derivation in (27), C_1 is the optimal output because no unchecked features remain on v or DP_{ext} . If no Agree took place, FULLINT and FC would be violated.

To conclude, we have derived pattern 3 and 4 of Table 3: If DP_{int} is 3rd person it bears no overt case marker, regardless of the features of DP_{ext} . There is no overt case marker because the optimal candidate of the v' -evaluation is a candidate without Maraudage and since the case marker reflects Maraudage, it cannot appear if DP_{int} is 3rd person.

We now turn to the derivations in which DP_{int} is atypical, i.e., 1st or 2nd person [BC]. We start with the evaluation of the v' -level. Since DP_{int} wants to check more features than v provides for it, Maraudage can apply to the probe feature [*B*] from set [*F*]_{ext} to set [*F*]_{int}.

⁷ See Heck & Müller (2003) for arguments that access to elements in the numeration is not another instance of look-ahead.

SCENARIO 2: DP_{int} is 1st/2nd person [BC]

(30) *Stage of the derivation* = v'

Input: $v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int}=[BC]$	FC	NOM	EOC
☞ $C_1: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int} [B\epsilon]$	*		
☞ $C_2: v \{ [BC*]_{ext}, [B\epsilon*]_{int} \}$ $DP_{int} [B\epsilon]$		*	
$C_3: v \{ [*BC*]_{ext}, [*C*]_{int} \}$ $DP_{int} [BC]$	**!		
☞ $C_4: \emptyset$			*

C_3 does not apply Agree at all, which leads to two violations of FC one of which is fatal. All other candidates violate the constraint tie only once and are thus optimal: C_1 checks only the feature [C] of DP_{int} , but its feature [B] remains unchecked because v does not provide [$*B*$] in the internal probe set. C_2 marauds [$*B*$] and can thereby avoid a violation of FC, but causes a violation of NOM.⁸ The empty output violates the EOC. The empty output can be ignored, it cannot be further expanded by structure building since it does not have any operation-inducing features; the two other optimal candidates can merge an external argument and can be further evaluated.

As a result of the evaluation at v' , C_1 without Maraudage and C_2 with Maraudage are the input for the optimization at the vP -level. In both a DP_{ext} of 1st/2nd or 3rd person can be merged. There are thus four possible derivations, but only two of them will converge and produce patterns 1 and 2 of Table 3. We continue with C_1 , in which no Maraudage applied and merge a 3rd person DP_{ext} in (31).

⁸ One might think of other candidates which represent further repair strategies beside Maraudage, e.g. deletion of the unchecked feature of a DP, or insertion of a probe feature counterpart on v , etc. I assume that these repair strategies are not available because the faithfulness constraints that militate against these repairs outrank the highest ranked constraint FULLINT.

SCENARIO 2.1: C₁ continued

(31) Stage of the derivation = vP, DP_{ext} is 3rd person [C]

Input: v {[*BC*] _{ext} , [*E*] _{int} } DP _{ext} =[C]	FULLINT	FC	NOM	EOC
C ₁ : v {[*BE*] _{ext} , [*E*] _{int} } DP _{ext} [E]	*!			
C ₂ : v {[*BC*] _{ext} , [*E*] _{int} } DP _{ext} [C]	*!***	*		
C ₃ : Ø				!

Since no Maraudage applied at the v'-stage, [*BC*] remains on v and must be checked. But DP_{ext} provides only [C] such that FULLINT is inevitably violated by the unchecked probe feature [*B*]. Only the empty output does not violate the highest ranked constraint FULLINT and is thus optimal. This means that there does not exist a pattern with a 1st/2nd person DP_{int}, a 3rd person DP_{ext} and without an overt case marker (since there is no Maraudage in this case). This is correct, cf. Table 3. Next, consider the case where a 1st/2nd person DP_{ext} is merged with C₁ of (30):

(32) Stage of the derivation = vP, DP_{ext} is 1st/2nd person [BC]

Input: v {[*BC*] _{ext} , [*E*] _{int} } DP _{ext} =[BC]	FULLINT	FC	NOM	EOC
C ₁ : v {[*BE*] _{ext} , [*E*] _{int} } DP _{ext} [BE]				
C ₂ : v {[*BE*] _{ext} , [*E*] _{int} } DP _{ext} [BE]	*!*	*		
C ₃ : v {[*BC*] _{ext} , [*E*] _{int} } DP _{ext} [BC]	*!***	**		
C ₃ : Ø				!*

In this case all remaining probe features of v and all the features of DP_{ext} can be checked. No constraint is violated as shown in C₁. In C₂ and C₃ only one or none of the features are checked and hence, FC and FULLINT are fatally violated. This derives pattern 1 in Table 3: In a context with a 1st/2nd person DP_{int} and a 1st/2nd person DP_{ext} there is no overt case marker because Maraudage did not apply.

The next two scenarios are those in which DP_{ext} is merged with the second optimal output C₂ of (30), the candidate in which Maraudage did apply. As a consequence of Maraudage, only the probe feature [*C*] remains in the set [*F*]_{ext} on v. We start with a 3rd person DP_{ext}.

SCENARIO 2.2: C₂ continued(33) *Stage of the derivation = vP, DP_{ext} is 3rd person [C]*

Input: v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} =[C]	FULLINT	FC	NOM	EOC
☞ C ₁ : v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} [€]				
C ₂ : v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} [C]	*!*	*		
C ₃ : Ø				!*

In C₁ v and DP_{ext} can check all of their features and hence none of the constraints is violated. No checking at all (C₂) or the empty output (C₃) violate FC and the EOC, respectively. The optimal candidate is thus C₁. This derives pattern 2 in Table 3: in the context 3rd person DP_{ext} and 1st/2nd DP_{int}, DP_{int} bears an overt case marker since Maraudage applied at the v'-level and there is a converging continuation of this derivation at the vP-level. The final case is the one in which C₂ of (30) and a 1st/2nd person DP_{ext} are merged.

(34) *Stage of the derivation = vP, DP_{ext} is 1st/2nd person [BC]*

Input: v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} =[BC]	FULLINT	FC	NOM	EOC
C ₁ : v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} [B€]	*!	*		
C ₂ : v {[* B C*] _{ext} , [* B C*] _{int} } DP _{ext} [BC]	*!***	**		
☞ C ₃ : Ø				!*

v provides only [*C*] after Maraudage, but DP_{ext} needs to check [BC]. Hence, FULLINT is inevitably violated by the unchecked feature [B] of DP_{ext}. As a consequence, the empty output becomes the optimal candidate. This means that there is no pattern in which both DP_{int} and DP_{ext} are 1st/2nd person and DP_{int} bears an overt case marker, cf. Table 3.

Thus, all the patterns in Table 3 and the non-existence of the other logically possible combinations of person features and overt vs. zero case marking in Yurok are derived. In a nutshell, the derivation goes as follows: Since Maraudage is optionally triggered if DP_{int} is high on the person hierarchy, namely 1st/2nd person, (i) there cannot be an overt case marker with a 3rd person DP_{int} and (ii) a converging derivation of vP does not necessarily result in an overt case marker because both a derivation with and one without Maraudage are optimal at the v'-level if DP_{int} is atypical. Only if DP_{ext} is such that it can check all of its own features and those of v, does a converging derivation arise. Otherwise,

the empty output wins, which amounts to a crash of the derivation, i.e., such a pattern does not exist.

Remember that the diacritic generated by Maraudage in the probe set [$*F*$]_{int} is transmitted via Agree to DP_{int} in Yurok. There it is realized by the vocabulary item given in (35) (it attaches only to singular arguments, hence the context restriction):⁹

- (35) *Case exponent in Yurok*
 -ac ↔ [◡] / [sg]

4.1.2. Umatilla Sahaptin

Another example of a binary split can be found in Umatilla Sahaptin (Penutian). DP_{ext} bears an ergative marker (glossed as INV.ERG) if DP_{int} is higher on the person scale in (36) than DP_{ext}.

- (36) *Person hierarchy in Umatilla Sahaptin*
 1st/2nd > 3rd

- (37) *GCS in Umatilla Sahaptin (Rigsby & Rude (1996, 676-677))*

- | | | |
|----|--|--------------|
| a. | iwínš i-tuxnana yáamaš-na
man 3NOM-shot mule.deer-OBJ
'The man shot a mule deer.' | 3rd sg > 3rd |
| b. | Ín=aš á-q'ínu-ša awínš-in-aman
1SG.NOM=1SG 3-see-IMPV men-DU-OBJ.PL
'I see the two men.' | 1st > 3rd |
| c. | iwínš-nim=nam i-q'ínu-ša
man-INV.ERG=2SG 3NOM-see-IMPV
'The man sees you.' | 3rd sg > 2nd |
| d. | iwínš-nim=naš i-wyánawi-yawan-a
man-INV.ERG=1SG 3SG-arrive-APPL-PST
'The man came to me / my place.' | 3rd sg > 1st |
| e. | Čáw=nam paamaná á-yk-ša?
NEG=2SG 3PL.OBJ 3-hear-IMPV
'Don't you hear them?' | 2nd > 3rd |

This leads to the following attested patterns:

⁹ Plural forms of a 1st/2nd person pronoun in Yurok never bear the accusative marker. This restriction cannot be handled by a second Maraudage operation that applies to number features because then we would expect to obtain restrictions on the number of DP_{ext} by the number of DP_{int}. But this is not the case. Therefore, I assume that -ac is a context-sensitive marker that can only be inserted in singular contexts.

Table 4. Person/case combinations in Umatilla Sahaptin

	person of DP _{ext}	case of DP _{ext}	person of DP _{int}	case of DP _{int}
Pattern 1:	1st/2nd	Abs	1st/2nd	Abs
Pattern 2:	3rd	Erg	1st/2nd	Abs
Pattern 3:	1st/2nd	Abs	3rd	Abs
Pattern 4:	3rd	Abs	3rd	Abs

The patterns in Umatilla Sahaptin are exactly the same as those in Yurok (compare Table 3) and the derivations are thus exactly the same; the only difference is the location of the split: an overt marker shows up on DP_{int} in Yurok, but on DP_{ext} in Umatilla Sahaptin. As was already discussed in 3.3, this difference is handled by a parameter which concerns the emergence of the diacritic when Maraudage takes place: it emerges in the set [**F**]_{int} in Yurok (attached to the marauded feature), but in the set [**F**]_{ext} (attached to the remaining features in the set which was affected by Maraudage) in Umatilla Sahaptin. It is then passed on to DP_{ext} in the latter and realized by an overt marker:

- (38) *Case exponent in Umatilla Sahaptin*¹⁰
 /nim/ ↔ [] / [sg]

4.2. Tripartite scale effects

There are also GCS languages in which overt case marking depends on a tripartite scale, namely Fore, Kashmiri and Awtuw. In this article, I concentrate on Fore (Trans-New Guinea) for ease of exposition. Case marking in Fore is driven by the animacy scale in (39). DP_{ext} bears an ergative suffix if it is lower on that scale than DP_{int}.

- (39) *Animacy hierarchy in Fore*
 human > animate > inanimate

- (40) *GCS in Fore (Scott (1978, 116))*

- a. Yagaa-wama wá aegúye
 pig-ERG man hit
 ‘The pig hits the man.’ *anim > hum*
- b. Yagaa wá aegúye
 pig man hit
 ‘The man hits (or kills) the pig.’ *hum > anim*

¹⁰ As in Yurok, the marker only attaches to singular arguments, hence the context restriction.

Table 5. Animacy/case combinations in Fore

	animacy of DP _{ext}	case of DP _{ext}	animacy of DP _{int}	case of DP _{int}
Pattern 1:	hum	Abs	hum	Abs
Pattern 2:	anim	Erg	hum	Abs
Pattern 3:	inanim	Erg	hum	Abs
Pattern 4:	hum	Abs	anim	Abs
Pattern 5:	anim	Abs	anim	Abs
Pattern 6:	inanim	Erg	anim	Abs
Pattern 7:	hum	Abs	inanim	Abs
Pattern 8:	anim	Abs	inanim	Abs
Pattern 9:	inanim	Abs	inanim	Abs

The only important difference between binary and tripartite scale effects for the analysis is the decomposition of features: three privative features are needed to encode a tripartite scale in order to distinguish the three steps on the hierarchy, but only two privative features are needed for a binary scale. As was done for person in Yurok and Umatilla Sahaptin, animacy is decomposed into privative features such that the value which is higher on the animacy scale has a superset of features compared to the value lower on the scale. In Fore, [C] is a general animacy feature (as opposed to person, number, etc.), [B] encodes [+animate], and [A] means [+human]. The following encodings result:

(41) *Representation of animacy features*

- a. [C] encodes inanimates.
- b. [BC] encodes animates.
- c. [ABC] encodes humans.

Again, *v* expects DP_{ext} to be higher on the scale than DP_{int}, namely that DP_{ext} is human and DP_{int} is inanimate. Hence, Maraudage which results in an overt case marker can potentially take place when DP_{int} is animate [BC] or human [ABC], cf. Table 5.

(42) *Lexical entry for v in Fore*

v [[*ABC*]_{ext}, [*C*]_{int}]

All other assumptions are exactly as in Yurok and Umatilla Sahaptin, especially the ranking of the constraints, i.e., Maraudage is optional. Under these assumptions the patterns in Fore can be derived in exactly the same way as in the two other languages, there are just more combinations of DP_{int} and DP_{ext} that could potentially be generated and Maraudage can apply to more than one feature if DP_{int} is human [ABC]. The gist of the analysis is again that since Maraudage need not and hence cannot apply with an inanimate DP_{int}, there will never be an overt case marker in these cases. Maraudage applies if DP_{int} is animate or

human. Regardless of whether Maraudage takes place or not at the v'-level, the derivation only converges at the vP-level when DP_{ext} is such that it matches exactly the remaining probe features of v, otherwise v and/or DP_{ext} have unchecked features and cause a fatal violation of FULLINT, which leads to the crash of the derivation.

4.3. Direction marking

In direction marking languages the occurrence of an overt verbal marker is driven by the same abstract pattern as the occurrence of the overt case marker in GCS languages: The verb bears an overt marker, the inverse marker, if DP_{int} is higher on a Silverstein scale than DP_{ext} . The verb in a direct context is usually zero-marked. Thus, direction marking differs from Global Case Splits only in the locus of the exponent – head-marking in direction marking languages vs. dependent-marking in GCS languages (Nichols (1986)). Hence, direction marking is another global argument encoding phenomenon which can be analysed in the same way as Global Case Splits (Zúñiga (2006); Drellishak (2008)).

To see the similarity more clearly, consider Nocte (Sino-Tibetan, Aissen (1999)): the direct, zero marked verb form is used if DP_{ext} is higher on the person scale in (43) than DP_{int} or if both are 3rd person (non-coreferent); the inverse marker *-h* is attached to the verb if DP_{int} is higher on the scale than DP_{ext} .

(43) *Person scale in Nocte*

1st > 2nd > 3rd

(44) *Person hierarchy effects in Nocte (Das Gupta (1971, 21))*

- | | | |
|----|---|-------------------------|
| a. | hetho-min
teach-1PL
'I will teach you(pl).' | <i>1st > 2nd</i> |
| b. | hetho-o
teach-2
'You will teach them.' | <i>2nd > 3rd</i> |
| c. | hetho-h-ang
teach-INV-1
'You/he will teach me.' | <i>2nd/3rd > 1st</i> |
| d. | hetho-h-o
teach-INV-2
'He will teach you.' | <i>3rd > 2nd</i> |

If person is decomposed as introduced in section 3.2, the following encodings of person features arises:

(45) *Person features in Nocte*

- a. 3rd person: [C]
- b. 2nd person: [BC]
- c. 1st person: [ABC]

If we now compare the abstract patterns of Nocte (inverse) with those of Fore (GCS), we can see that they are identical (the gaps in Nocte are due to the fact that person is the decisive feature in Nocte – the gaps are reflexive contexts; since the relevant feature in Fore is animacy, no such gaps arise):

(46) *Patterns of Fore and Nocte*

Fore		Nocte	
ergative	∅	inverse	∅
[BC]–[ABC]	[ABC]–[ABC]	[BC]–[ABC]	–
[C]–[ABC]	[ABC]–[BC]	[C]–[ABC]	[ABC]–[BC]
[C]–[BC]	[BC]–[BC]	[C]–[BC]	–
	[ABC]–[C]		[ABC]–[C]
	[BC]–[C]		[BC]–[C]
	[C]–[C]		[C]–[C]

Thus, these patterns should be derived in the same way, although one shows dependent-marking and the other head-marking. Under the Maraudage analysis developed in this article, direction marking in Nocte can be derived in the same local way as the Global Case Split in Fore. Note that no further assumptions are necessary to derive direction marking, the pattern is even expected under the analysis given that the diacritic which shows that Maraudage has taken place is generated on *v*, a verbal projection. If DP_{int} is atypical, Maraudage can apply. Depending on the properties of DP_{ext} , a derivation in which Maraudage has applied can become optimal. The diacritic is then realized postsyntactically on *v* instead of on an argument DP.¹¹ The difference in locus between GCS and direction marking languages can be modeled by context restrictions on the relevant VI which make them category-sensitive:

- (47) a. /X/ ↔ [] / _ [v] *inverse marker*
- b. /X/ ↔ [] / _ [D] *case marker*

¹¹ It is of no importance whether in direction marking languages the diacritic is also transmitted to a DP via Agree or not. Either it is not copied to a DP or it is copied just as in GCS languages but it is simply not spelled-out on the DP. Since there are languages which have direction marking and GCS simultaneously, e.g. Arizona Tewa (Kroskrity (1978; 1985); Zúñiga (2006)), the second option seems preferable.

5. Languages without Global Case Splits

The analysis developed for languages with Global Case Splits might seem to be designed for the relatively small number of languages with GCS. In this section I show that the analysis can also handle languages with Local Case Splits (LCS) and languages without case splits. Finally, I address how Burzio's generalization can be derived from the Maraudage approach.

5.1. Local Case Splits

The main difference between Global and Local Case Splits is that in the latter case marking solely depends on the properties of DP_{int} and not also on those of the coargument. To model this in the present analysis, Maraudage must be obligatory if it can apply, i.e., whenever DP_{int} is atypical. Hence, FC must outrank NOM, $FC \gg NOM$. As a result, only one candidate will be optimal when the v' -level is evaluated: if DP_{int} is higher on the scale than expected, the candidate that applies Maraudage is optimal and it will be the input for optimization at the vP -level. There will thus always be an overt case marker with an internal argument which is high on a Silverstein scale. The ranking in languages with Local Case Splits is given in (48):

- (48) *Ranking for LCS*
 $EOC \gg FULLINT \gg FC \gg NOM$

Another contrast between languages with GCS and LCS is that the constraint EOC must be the highest ranked in the latter. The reason is the context in which DP_{ext} and DP_{int} are both high on a scale. Take, for example, a language that is like Yurok in that the split depends on a binary scale, but it has a local split. v expects DP_{int} to be low on the scale: $v \{ [*BC*]_{ext} [*C*]_{int} \}$. If DP_{int} is high on the binary scale, viz. [BC], Maraudage must apply under the ranking $FC \gg NOM$. The result is that v provides only $[*C*]_{ext}$ for the external argument. If, however, DP_{ext} is also high on the scale, viz. [BC], its feature [B] cannot be checked because $[*B*]$ has been marauded and checked by DP_{int} and hence, FULLINT and FC must be violated. If the EOC was ranked as in GCS languages, this would wrongly predict that the empty output is the optimal candidate in such a situation. But in LCS languages, if there is an overt case marker, i.e., if Maraudage has applied, a DP_{ext} with any properties can be merged – it will always lead to an attested pattern and hence, the derivation must converge (cf. Hebrew and Tauya below). This means that a violation of FULLINT and FC by an unchecked feature of a DP cannot be fatal in LCS languages. Therefore, the EOC is the highest ranked constraint, the empty output can never be optimal.

With respect to the locus of the diacritic, the same variation arises as in GCS languages: the diacritic indicating Maraudage can be generated in $[*F*]_{ext}$ or in $[*F*]_{int}$ and is passed on to DP_{ext} and DP_{int} , respectively. The latter case is the most common local split in which the same argument on whose properties the split depends exhibits the split. An example for such an accusative case marking pattern is Hebrew as shown in (2). In Hebrew, DP_{int} bears a case marker if it is high on the binary definiteness scale, viz. definite (a pronoun, a name or a definite noun). In any other configuration, it is zero-marked (nominative):

- (49) *Definiteness hierarchy in Hebrew*
 definite \succ indefinite

Table 6. *Definiteness/case combinations in Hebrew*

	DP_{ext} case of DP_{ext}	DP_{int} case of DP_{int}
Pattern 1:	def/indef Nom	def Acc
Pattern 2:	def/indef Nom	indef Nom

What has to be done is to decompose definiteness into privative features: [C] is a general definiteness feature (as opposed to person, animacy, etc.) and [B] means [+definite]. Hence, the following feature bundles for definites and indefinites arise:

- (50) *Representation of definiteness features*
 a. [C] encodes an indefinite referent
 b. [BC] encodes a definite referent

v in Hebrew also expects DP_{int} to be lower on the scale than DP_{ext} :

- (51) *v in Hebrew*
 v $[[*BC*]_{ext}, [*C*]_{int}]$

Under the ranking in (48) the patterns in Table 6 are derived. The diacritic is generated in $[*F*]_{int}$ and transferred to DP_{int} under Agree where it is spelled-out as *?et*.

There are also LCS languages with an ergative case marking pattern, i.e., in which the split depends on the properties of DP_{int} but the case marker alternation shows up on DP_{ext} . Tauya (Trans-New Guinea) is such a language. In Tauya, DP_{ext} has an overt marker if DP_{int} is high on the binary animacy hierarchy, viz. if it is human.

(52) *Animacy hierarchy in Tauya*
 human \succ non-human

(53) *GCS in Tauya (MacDonald (1990, 120-121 & 316))*

- a. ya-ni/* \emptyset fanu \emptyset -yau-e-ʔa
 1SG-ERG/*ABS man 3SG-see-1/2-IND
 ‘I saw the man.’ *hum > hum*
- b. ya- \emptyset pai yau-e-ʔa
 1SG-ABS pig see-1/2-IND
 ‘I saw the pig.’ *hum > non-hum*

Table 7. *Animacy/case combinations in Tauya*

	DP _{ext}	case of DP _{ext}	DP _{int}	case of DP _{int}
Pattern 1:	hum/non-hum	Erg	hum	Abs
Pattern 2:	hum/non-hum	Abs	non-hum	Abs

The split can be derived in the present system if animacy is decomposed as in (54) and under the ranking in (48).

(54) *Representation of animacy features*

- a. [C] is a general animacy feature
- b. [B] means [+human]
- c. [C] encodes a non-human referent
- d. [BC] encodes a human referent

(55) *v in Tauya*

v [[*BC*]_{ext}, [*C*]_{int}]

The diacritic indicating Maraudage is generated in the set [*F*]_{ext} on v and transmitted to DP_{ext} via Agree where it is realized by an overt marker. This gives rise to an ergative pattern of case marking.

5.2. Languages without case splits

In languages without case splits an argument A always shows the same case marker, regardless of the nature of DP_{int}. In German, for example, DP_{int} of a transitive verb always bears accusative case (except when a verb assigns inherent case which overwrites the default accusative). If Maraudage leads to overt case marking, it must apply in every derivation in these languages. To guarantee that this is enforced, languages without case splits have to be treated like languages with Local Case Splits, i.e., Maraudage must be obligatory if possible, FC \gg

NOM. In order to ensure that Maraudage can apply even with typical internal arguments which are low on a scale, the probe set $[*F*]_{int}$ on v must be empty:

$$(56) \quad v \{ [* (A)(B)C*]_{ext}, [*F*]_{int} \}$$

The external probe set must at least contain the feature which is part of every value of a category, i.e., [C] (it can contain more but this does not matter for the core of the analysis). Note that this representation also matches the requirement in GCS and LCS languages that v provides for Agree with DP_{int} a subset of features of the features provided for Agree with DP_{ext} . The consequence of the empty internal probe set is that any DP_{int} can trigger Maraudage because DP_{int} always contains at least the feature [C]. Given the ranking $FC \gg NOM$ Maraudage must then apply. As in LCS languages, the properties of DP_{ext} are of no importance: FULLINT and FC are non-fatally violable under the ranking in (48). Depending on where the diacritic that indicates Maraudage is generated (in the set $[*F*]_{int}$ or in the set $[*F*]_{ext}$), an ergative or an accusative pattern of case marking arises.

5.3. Burzio's Generalization

An interesting consequence of the present approach is that Burzio's Generalization can be accounted for. Reformulated in modern terms, Burzio (1986) states that only the v which selects an external argument can assign accusative. Burzio's generalization is on abstract case, but in the present system there is no abstract case feature (cf. footnote 6), hence the correlation can only hold for morphological case. The derivation goes as follows: Morphological accusative case is always the indicator of the operation Maraudage. Maraudage is only possible if there are two probe feature sets on v . But since in intransitive contexts it is generally assumed that there can be only one probe for the single argument if the derivation is to converge ($v \{ [*F*] \}$), Maraudage is excluded: there is no other probe feature set from which features could be displaced. Consequently, no diacritic is generated and there can be no accusative marking. The same holds for transitive verbs which are passivized. A necessary step for passivization is argument reduction which can be brought about by deletion of $[\bullet D \bullet]$ on v in the present approach. Now, if this includes deletion of one of the probe sets on v since passivization is a detransitivizing operation and intransitive v has only one probe set, then again only a single probe set remains on v after argument reduction and Maraudage is precluded.

6. Conclusion

In this paper I have shown how Global Case Splits can be derived in a local and cyclic way. GCSs have rarely been addressed in the literature although they are a challenge for a cyclic derivational syntax: they either require look-ahead or counter-cyclic case assignment under standard minimalist assumptions about structure building and case assignment. I proposed that the data should be looked at from a new perspective: The features of DP_{ext} depend on those of DP_{int} . Whenever DP_{int} is atypical (viz. high on a scale), the repair operation *Maraudage* can apply: it steals features on v provided for DP_{ext} in order to use them for Agree with DP_{int} . As only the features of DP_{int} are relevant for *Maraudage*, a local derivation is possible. At no point in the derivation is the information about both arguments of a transitive verb accessible. Overt case marking is analysed as a reflex of *Maraudage*. The typology of global and local case marking strategies is derived by two parameters: (i) *Maraudage* is obligatory (FC \gg NOM) or optional (FC \gg NOM) and (ii) the diacritic is generated in the external or the internal probe set on v , which accounts for ergative vs. accusative patterns of case marking. Furthermore, it was shown that the analysis carries over to another global argument encoding pattern, namely inverse marking, to languages without case splits, and finally that it can derive Burzio's Generalization.

Bibliography

- Adger, David & Daniel Harbour (2007): 'Syntax and Syncretisms of the Person Case Constraint', *Syntax* 10, 2–37.
- Aissen, Judith (1999): 'Markedness and Subject Choice in Optimality Theory', *Natural Language and Linguistic Theory* 17, 673–711.
- Aissen, Judith (2003): 'Differential Object Marking: Iconicity vs. Economy', *Natural Language and Linguistic Theory* 21, 435–483.
- Anagnostopoulou, Elena (2003): *The Syntax of Ditransitives: Evidence from Clitics*. Mouton de Gruyter, Berlin.
- Béjar, Susana (2003): *Phi-Syntax: A Theory of Agreement*. PhD thesis, University of Toronto.
- Béjar, Susana & Milan Režáč (2009): 'Cyclic Agree', *Linguistic Inquiry* 40, 35–73.
- Bossong, Georg (1985): *Differenzielle Objektmarkierung in den neuiranischen Sprachen*. Narr, Tübingen.
- Burzio, Luigi (1986): *Italian Syntax*. Kluwer, Dordrecht.
- Chomsky, Noam (1965): *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.
- Chomsky, Noam (1973): Conditions on Transformations. In: S. Anderson & P. Kiparsky, eds., *A Festschrift for Morris Halle*. Academic Press, New York, pp. 232–286.
- Chomsky, Noam (1995): *The Minimalist Program*. MIT Press, Cambridge, Mass.
- Chomsky, Noam (2000): Minimalist Inquiries: The Framework. In: R. Martin, D. Michaels & J. Uriagereka, eds., *Step by Step*. MIT Press, Cambridge, Mass., pp. 89–155.
- Chomsky, Noam (2001): Derivation by Phase. In: M. Kenstowicz, ed., *Ken Hale. A Life in Language*. MIT Press, Cambridge, Mass., pp. 1–52.
- Comrie, Bernard (1979): 'Definite and Animate Direct Objects: A Natural Class', *Linguistica silesiana* 3, 13–21.

- Das Gupta, Kamalesh (1971): *An Introduction to the Nocte Language*. North-East Frontier Agency, Shillong.
- De Hoop, Helen & Andrej L. Malchukov (2008): 'Case-Marking Strategies', *Linguistic Inquiry* 39, 565–587.
- Drellishak, Scott (2008): Complex Case Phenomena in the Grammar Matrix. In: S. Müller, ed., *The Proceedings of the 15th International Conference on Head-Driven Phrase Structure Grammar*. CSLI Publications, Stanford, pp. 67–86. Available at <http://csli-publications.stanford.edu/HPSG/9/>.
- Feldman, Harry (1986): *A Grammar of Awtuw*. Australian National University, Canberra.
- Georgi, Doreen (2009): Local Modelling of Global Case Splits. Master's thesis, University of Leipzig.
- Hale, Ken (1972): A New Perspective on American Indian Linguistics. In: A. Ortiz, ed., *New Perspectives on the Pueblos*. University of New Mexico Press, Albuquerque, pp. 87–103.
- Halle, Morris & Alec Marantz (1993): Distributed Morphology and the Pieces of Inflection. In: K. Hale & S. J. Keyser, eds., *The View from Building 20*. MIT Press, Cambridge, Mass., pp. 111–176.
- Halle, Morris & Alec Marantz (1994): Some Key Features of Distributed Morphology. In: A. Carnie, H. Harley & T. Bures, eds., *Papers on Phonology and Morphology*. Vol. 21 of *MIT Working Papers in Linguistics*, MITWPL, Cambridge, Mass., pp. 275–288.
- Harbour, Daniel (2008): A Feature Calculus for Silverstein Hierarchies. Handout of a talk given at the Workshop on Scales, University of Leipzig, March 2008.
- Harley, Heidi & Elisabeth Ritter (2002): 'Person and Number in Pronouns: A Feature-Geometric Analysis', *Language* 78, 482–526.
- Harley, Heidi & Rolf Noyer (1999): 'Distributed Morphology', *GLoT International* 4/4, 3–9.
- Heck, Fabian & Gereon Müller (2003): 'Derivational Optimization of Wh-Movement', *Linguistic Analysis* 33, 97–148.
- Heck, Fabian & Gereon Müller (2007): Extremely Local Optimization. In: E. Brainbridge & B. Agbayani, eds., *Proceedings of WECOL 26*. California State University, Fresno, pp. 170–183.
- Heck, Fabian & Marc Richards (2010): 'A Probe-Goal Approach to Agreement and Non-Incorporation Restrictions in Southern Tiwa', *Natural Language and Linguistic Theory* 28, 681–721.
- Keine, Stefan (2010): *Case and Agreement from Fringe to Core. A Minimalist Approach*. Linguistische Arbeiten 536, de Gruyter, Berlin.
- Keine, Stefan & Gereon Müller (2010): Non-zero/Non-zero Alternations in Differential Object Marking. In: P. Brandt & M. García García, eds., *Transitivity. Form, Meaning, Acquisition, and Processing*. John Benjamins, Amsterdam/Philadelphia, pp. 119–140. *Linguistik Aktuell/Linguistics Today* 166.
- Kroskrity, Paul V. (1978): 'Aspects of Syntactic and Semantic Variation Within the Arizona Tewa Speech Community', *Anthropological Linguistics* 20, 235–257.
- Kroskrity, Paul V. (1985): 'A Holistic Understanding of Arizona Tewa Passives', *Language* 61, 306–328.
- Lazard, Gilbert (1984): Actance Variations and Categories of the Object. In: F. Plank, ed., *Objects: Towards a Theory of Grammatical Relations*. Academic Press, London, pp. 249–289.
- Legate, Julie Anne (2008): 'Morphological and Abstract Case', *Linguistic Inquiry* 39, 55–101.
- MacDonald, Lorna (1990): *A Grammar of Taya*. Mouton de Gruyter, Berlin.
- Maslova, Elena (2003): *A Grammar of Kolyma Yukaghir*. Mouton Grammar Library, de Gruyter, Berlin, New York.
- Müller, Gereon (2000): *Elemente der optimalitätstheoretischen Syntax*. Stauffenburg, Tübingen.
- Nichols, Johanna (1986): 'Head-Marking and Dependent-Marking Grammar', *Language* 62, 56–119.
- Noyer, Rolf (1998): Impoverishment Theory and Morphosyntactic Markedness. In: S. Lapointe, D. Brentari & P. Farrell, eds., *Morphology and its Relation to Phonology and Syntax*. CSLI, Palo Alto, pp. 264–285.

- Prince, Alan & Paul Smolensky (1993): Optimality Theory. Constraint Interaction in Generative Grammar. Book ms., Rutgers University.
- Řezáč, Milan (2008): 'The Syntax of Eccentric Agreement: The Person Case Constraint and Absolute Displacement in Basque', *Natural Language and Linguistic Theory* 26, 61–106.
- Richards, Marc (2008): Quirky Expletives. In: R. d'Alessandro, G. H. Hrafnbjargason & S. Fischer, eds., *Agreement Restrictions*. Mouton de Gruyter, Berlin, pp. 181–213.
- Rigsby, Bruce & Noel Rude (1996): Sketch of Sahaptin, a Sahaptin language. In: I. Goddard, ed., *Handbook of North American Indians: Languages*. Smithsonian Institution, Washington, pp. 666–692.
- Robins, Robert H. (1958): *The Yurok Language: Grammar, Texts, Lexicon*. University of California Press, Berkeley.
- Scott, Graham (1978): *The Fore Language of Papua New Guinea*. Pacific Linguistics (PL-B47), Canberra.
- Silverstein, Michael (1976): Hierarchy of Features and Ergativity. In: R. Dixon, ed., *Grammatical Categories in Australian Languages*. Australian Institute of Aboriginal Studies, Canberra, pp. 112–171.
- Sternefeld, Wolfgang (2006): *Syntax*. Stauffenburg, Tübingen. Two volumes.
- Wali, Kashi & Omkar N. Koul (1997): *Kashmiri. A Cognitive-Descriptive Grammar*. Routledge, London.
- Zúñiga, Fernando (2006): *Deixis and Alignment. Inverse Systems in Indigenous Languages of the Americas*. Vol. 70 of *Typological Studies in Language*, John Benjamins, Amsterdam.

*Institut für Linguistik
Universität Leipzig*