Optimality-Theoretic Syntax
Gereon Müller (Universität Leipzig), WiSe 2012/2013

Contents

1. Straßenverkehrsordnung .................................................. 2
2. Model of Grammar .................................................................. 3
3. Evidence for OT Analyses in Syntax ...................................... 7
    3.1 Constraint Conflict ............................................................. 8
    3.2 Repair Phenomena ............................................................. 9
        3.2.1 Do-Support .............................................................. 9
        3.2.2 Resumptive Pronouns ................................................. 13
        3.2.3 Other Repair Phenomena ............................................ 14
    3.3 Default Contexts .............................................................. 15
4. Problems for OT Analyses in Syntax ...................................... 18
    4.1 Complexity ....................................................................... 18
    4.2 Ineffability (Absolute Ungrammaticality) ......................... 19
        4.2.1 The Generator ........................................................ 20
        4.2.2 Empty Outputs ......................................................... 20
        4.2.3 Bad Winners ............................................................ 20
        4.2.4 Repair ..................................................................... 21
        4.2.5 Neutralization ......................................................... 22
        4.3 Optionality .................................................................. 23
            4.3.1 Pseudo-Optionality ............................................... 24
            4.3.2 True Optionality .................................................... 24
            4.3.3 Ties .................................................................... 25
            4.3.4 Stochastic Optimality Theory ................................ 26
5. Optimization Domains ........................................................... 28
    5.1 Introduction .................................................................... 28
    5.2 Clauses as optimization domains ...................................... 30
    5.3 Derivational Steps as Optimization Domains ..................... 31
        5.3.1 Constraints, Features, and Operations ..................... 32
        5.3.2 Empirical Evidence for Extremely Local Optimization 33
            1: Argument Encoding ................................................. 33
        5.3.3 Prenominal Dative Possessors in German .................. 37
        5.3.4 Conclusion ............................................................. 39
6. Ubiquity of Optimization ..................................................... 40
    6.1 Background ..................................................................... 40
    6.2 Hidden Optimization ...................................................... 40
    6.3 Optimization in the Minimalist Program ......................... 42
7. Abstractness of Optimality-Theoretic Syntax .......................... 44

1. Straßenverkehrsordnung

(1) Vorfahrtsregeln
   a. V(ERKEHRS)-POL(IZIST) (StVO § 36):
      Die Zeichen und Weisungen der Polizeibeamten auf der Kreuzung sind zu befolgen.
      ⇒ ("Sie gehen den Anordnungen der ortsfest angetrackten Verkehrsteilnehmer vor.")
   b. BL(AULICHT) - EIN(SATZHORN) (StVO §§ 35, 38):
      Fahrzeuge des Rettungsdienstes und der Polizei dürfen blaues Blinklicht zusammen mit einem Einsatzhorn verwenden; dies ordnet an: "Alle übrigen Verkehrsteilnehmer haben sofort freie Bahn zu schaffen." ⇒
   c. L(ICH) - ZEI(chen) (StVO § 37):
      An Kreuzungen bzw. Einmündungen bedeuten Grün: "Der Verkehr ist freigegeben"; Gelb: "Vor der Kreuzung auf das nächste Zeichen"; Rot: "Halt vor der Kreuzung". ⇒ ("Lichtzeichen geben Vorrangregeln und vorrangsgreifenden Verkehrsschildern ... vor.")
   d. V(ERKEHRS)-ZEI(chen) (a) (StVO § 39):
      Verkehrsschilden auf einem Fahrzeug ist Folge zu leisten. Sie gelten auch, wenn das Fahrzeug sich bewegt. ⇒ ("Sie gehen den Anordnungen der ortsfest angetrackten Verkehrsteilnehmer vor.")
   e. V(ERKEHRS)-ZEI(chen) (b) (StVO § 39):
      Ortstenfesten Verkehrsschilden ist Folge zu leisten. ⇒ ("Regelungen durch Verkehrsschilden geben den allgemeinen Vorrangregeln vor.")
   f. S(TRASSE) V(or) F(eldweg) (StVO § 8):
      Fahrzeuge, die aus einem Feld- oder Waldweg auf eine andere Straße kommen, haben Vorfahrt zu gewähren. ⇒
   g. R(ECHTS) V(or) L(INKS) (StVO § 8):
      An Kreuzungen und Einmündungen hat die Vorfahrt, wer von rechts kommt.
T₁: BL-EIN \gg Li-Zei

<table>
<thead>
<tr>
<th>I: A(L, rot, []), B(R, V, grün, [)]</th>
<th>V-POL</th>
<th>BL-EIN</th>
<th>Li-Zei</th>
<th>V-Zei</th>
<th>SVF</th>
<th>RVL</th>
</tr>
</thead>
<tbody>
<tr>
<td>O₁: A vor B</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O₂: B vor A</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₃: A und B</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

T₂: V-POL \gg andere Regeln

<table>
<thead>
<tr>
<th>I: A(L, V, rot, []), B(R, V, grün, [)]</th>
<th>V-POL</th>
<th>BL-EIN</th>
<th>Li-Zei</th>
<th>V-Zei</th>
<th>SVF</th>
<th>RVL</th>
</tr>
</thead>
<tbody>
<tr>
<td>O₁: A vor B</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O₂: B vor A</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₃: A und B</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bemerkung:
- ein durch den Verkehrsbeauftragten erteiltes Haltegebot
- durch den Verkehrsbeauftragten erteilte Fahrtfreigabe

T₃: V-POL \gg V-Real

<table>
<thead>
<tr>
<th>I: A(L, V, rot, []), B(R, V, grün, [)]</th>
<th>V-POL</th>
<th>V-Real</th>
<th>BL-EIN</th>
<th>Li-Zei</th>
<th>V-Zei</th>
<th>SVF</th>
<th>RVL</th>
</tr>
</thead>
<tbody>
<tr>
<td>O₁: A vor B</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₂: B vor A</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₃: A und B</td>
<td>*!</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Model of Grammar

Optimality Theory (OT) has been developed since the early nineties, by Alan Prince, Paul Smolensky, John McCarthy and others. At first, the focus was mainly on phonology, but the approach has since been extended to morphology, syntax, semantics, and pragmatics. The most comprehensive (and best) exposition of the theory is still Prince & Smolensky (1993; 2004). Early groundbreaking work in syntax includes Grimshaw (1997), Pesetsky (1993, 2004). Early groundbreaking work in syntax includes Grimshaw (1997), Pesetsky (1993, 2004), and Legendre, Smolensky & Wilson (1998). Introductions include Kager (1999) (with little material on syntax). Müller (2000b) (in German), Legendre (2001), and McCarthy (2002) (with quite a bit on syntax). (3) Basic assumptions of grammatical theories that do without competition (e.g., Chomsky's (1981) theory of Government and Binding):

- a. Not all constraints are universal (parameters, language-specific filters – but cf. ‘third-factor’ meta-constraints on constraints in recent work in the minimalist program, as in Chomsky (2007, 2008)).
- b. Constraints cannot be violated.
- c. Constraints are not ranked (all are equally important – but cf., e.g., Subjacency vs. ECP violations).
- d. The wellformedness of a linguistic expression (e.g., a sentence) LE can fully be determined on the basis of LE’s internal properties. External factors (i.e., the properties of other LE’s) are irrelevant.

(4) Optimality:

A candidate \( C_i \) is optimal with respect to some constraint ranking \( < Con_1 \gg Con_2 \gg ... \gg Con_n > \) iff there is no other candidate \( C_j \) in the same candidate set that has a better constraint profile.

(5) Constraint profile:

\( C_j \) has a better constraint profile than \( C_i \) if there is a constraint \( Con_k \) such that (i) and (ii) hold:

- a. \( C_j \) satisfies \( Con_k \) better than \( Con_i \).
- b. There is no constraint \( Con_i \) that is ranked higher than \( Con_k \), and for which \( C_i \) and \( C_j \) differ.

Note: \( C_j \) satisfies a constraint \( Con \) better than \( C_i \) if \( C_j \) violates \( Con \) less often than \( C_i \). This includes the case where \( C_j \) does not violate \( Con \) at all, whereas \( C_i \) does.

(6) Candidate set (possible definitions in syntax):

Two candidates are in the same candidate set if:

- a. they have the same content words
- b. they have the same words (Chomsky’s (1999) numeration)
- c. they have the same meaning
- d. they have the same words and the same meaning
- e. they have the same words and the same meaning
Structure of an optimality-theoretic syntax component

Generator

I₁

⇒

O₁

⇒

O₂

⇒

O₃

⇒

...

⇒

Oₙ

input: numeration? pred/arg structure? nothing?

Output candidates

part (i) of the grammar: inviolable, unordered constraints; simple standard grammar

part (ii) of the grammar: violable, ranked, universal constraints; genuine OT grammar

H(armony)-Eval(uation)

Oₙ

optimal output: well-formed candidate

(7) The input/output distinction:

a. Standard OT assumption: Gen creates outputs candidates on the basis of an input; i.e., inputs also define the candidate set.

b. Outputs differ from their underlying input in various ways (giving rise to faithfulness violations; see below), but inputs are standardly assumed to be of roughly the same type as outputs (e.g., UROs in phonology), and may even be identical. This seems hardly tenable for syntax. (If outputs are syntactic structures, and structures are generated by Gen, then where does the input structure come from if inputs are also syntactic structures?)

c. Consequently, it is completely unclear what the input in syntax should look like; perhaps there is no input in syntax at all. See Heck et al. (2002), where it is argued that the two basic motivations for inputs – (a) defining candidate sets and (b) providing information for faithfulness constraints – are either unavailable (a) or irrelevant (b) in syntax (because syntax, unlike, phonology, is an information-preserving system, with, e.g., subcategorization information present on a verb throughout the derivation).

d. Anyway, from now on: candidates = outputs.

Types of constraints:

Standardly, two basic types of H-Eval constraints can be distinguished that often give rise to conflicts:

- Faithfulness constraints demand that input and output are identical with respect to some property (no addition of items in the output: Dep; no deletion of items in the output: Max; no change of items in the output: Ident).
- Markedness constraints impose requirements on outputs that may necessitate a deviation from the input.

Note:

Optimality-theoretic competitions are often illustrated by tables, so-called tableaux.

Type: The basic principle

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>O₁</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₂</td>
<td></td>
<td>**</td>
<td></td>
</tr>
<tr>
<td>O₃</td>
<td></td>
<td></td>
<td>*!</td>
</tr>
<tr>
<td>O₄</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₅</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Generalization:

The optimal output is the candidate that has its first star furthest to the right in a tableau.

(8) Cross-linguistic variation:

Assumption (not made in minimalist approaches, but virtually everywhere else): Languages differ with respect to their grammars. Grammatical differences between languages = parametrization. Parametrization in optimality theory: constraint reranking.

Note:

Optimality theory was developed out of so-called “harmonic grammar” approaches → theory of neural networks. (Further reading: Prince & Smolensky (2004, ch. 10), Smolensky & Legendre (2006, part I). Main innovation: Quality before quantity; no number of violations of a lower-ranked constraint can outweigh a single violation of a higher-ranked constraint.

Caveat:

OT has introduced a means to undermine the irrelevance of constraint violation quantity as
### T_7: Parametrization

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>O_2</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_3</td>
<td>****</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O_4</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_5</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

**T_8: Irrelevance of constraint violation numbers as such**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>O_1</td>
<td>****</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O_2</td>
<td>*!</td>
<td></td>
<td>****</td>
</tr>
<tr>
<td>O_3</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_4</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_5</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

### T_9: A consequence of reflexive local conjunction

Input: John, bought, which, book, v, T, C

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>O_1</td>
<td>*!</td>
<td></td>
<td>****</td>
</tr>
<tr>
<td>O_2</td>
<td>*!</td>
<td></td>
<td>********</td>
</tr>
<tr>
<td>O_3</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_4</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>O_5</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

3. Evidence for OT Analyses in Syntax

**Evidence for OT analyses:**

- constraint conflict
- repair phenomena
- cross-linguistic variation by constraint re-ranking

### 3.1 Constraint Conflict

**Profile of the empirical evidence:**

The facts show that two general and far-reaching constraints are well motivated, independently of each other. However, in some contexts the two constraints may end up being in conflict, with the evidence suggesting that one may selectively, and systematically, be violated in favour of the other. In standard approaches to grammar, this state of affairs automatically gives rise to an undesirable consequence: One of the two constraints must be abandoned; or there has to be an explicit exception clause in the definition of one of the constraints; or the application of one of the two constraints has to be relegated to some other (typically more abstract) level of representation; etc. In an OT grammar, the constraint conflict can be systematically resolved by constraint ranking.

(10) **Wh-movement in English:**

a. I don’t know [CP which book John bought]

b. * I don’t know [CP John bought which book]

(11) **Two plausible constraints:**

a. **Wh-Criterion (Wh-Crit):**

   Wh-items are in SpecC_{wh}.

b. **θ-Assignment (θ-Assign):**

   Internal arguments of V are c-commanded by V.

**Consequence for standard models of grammar:**

Either θ-Assign does not hold; or the constraint is enriched by an exception clause (“does not hold for wh-items”); or both constraints hold, but not at the same level of representation (Wh-Crit may hold for surface representations or S-structure, θ-Assign may hold for an abstract level of predicate argument structure or D-structure).

**Consequence in OT:**

Both constraints hold, but they are ranked.

(12) **Ranking:**

Wh-Crit ≫ θ-Assign

### T_{10}: Simple wh-question formation in English

Input: John, bought, which, book, v, T, C_{[+wh]}

<table>
<thead>
<tr>
<th></th>
<th>O_1: ... which book John bought</th>
<th>Wh-Crit</th>
<th>θ-Assign</th>
</tr>
</thead>
<tbody>
<tr>
<td>O_2: ... John bought which book</td>
<td>!</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>
Movement may be assumed leave a trace (t) or a copy. The role of θ-ASSIGN can then be taken over by the general constraint Economy (Grimshaw (1997), Legendre, Smolensky & Wilson (1998), Ackema & Neeleman (1998), etc.).

(13) **Economy:**
Traces (copies) are prohibited.

*Note:*
Arguably, (13) can (and should) be derived from more general constraints and their interaction (see Grimshaw (2001; 2006); Steddy & Samek-Lodovici (2009) for an application of the underlying logic to universal constraints on DP-internal order of D, Num, A and N).

**Observation:**
(i) All syntactic constituents violate alignment constraints (given dichotomies like HEAD-LEFT/HEAD-RIGHT, COMP-LEFT/COMP-RIGHT). More structure implies more violations of alignment constraints. Movement is structure-building (Grimshaw (2001)).
(ii) All chains are trivial (single-membered) in the input. Movement gives rise to non-trivial (multi-membered) chains. This implies a violation of faithfulness (IDENT/UNIQUENESS; see Grimshaw (2006)).

**To sum up:**
In conclusion, whether UG constraints conflict or not is an empirical issue. If they do, and they do appear to do so, a formally precise theory of their interaction becomes necessary for a proper understanding of grammar because simultaneous satisfaction of all constraints ceases to be a viable definition of grammaticality.

Samek-Lodovici (2006, 94)

### 3.2 Repair Phenomena

**Profile of the empirical evidence:**
The facts suggest that some grammatical complex LE exhibits properties that are not normally permitted in the grammar. It seems that, in the case at hand, these properties are permitted as a last resort (given that all alternatives are even worse, in a sense to be made precise).

#### 3.2.1 Do-Support

**Lit.:**
Chomsky (1957; 1991)

(14) *Do*-Einsetzung im Englischen bei Negation:
   a. *Mary not left
   b. Mary did not leave

(15) Keine *do*-Einsetzung im Englischen bei fehlender Negation:
   a. Mary left
   b. *Mary did leave

(16) Lösung in Chomsky (1957, 39 & 62):
   a. **Affix Hopping**
      Let A\(f\) stand for any of the affixes past, S, en, ing. Let v stand for any M(odal) or V(erb), or have or be (i.e., for any non-affix in the phrase Verb). Then:
      \[ A\!f + v \rightarrow v + A\!f\# \],
      where # is interpreted as word boundary.
      Replace + by # except in the context v – A\(f\). Insert # initially and finally.
   b. **T_not**:
      T\(not\) adds not after the second segment of the string.
   c. **Do Support**:
      #A\(f\) → #do+ A\(f\).
   d. **Assumption:** T\(not\) applies before Affix Hopping (bleeding), Do Support applies after Affix Hopping (counter-feeding).

Intuition: Affix Hopping wird blockiert bei Nicht-Adjazenz von A\(f\) und v (nach Negations-einsetzung). In diesem Fall, und nur in diesem, erfolgt dann Do Support (weil A\(f\) am linken Rand geblieben ist).

(17) Keine *do*-Einsetzung bei Negation und have:
   a. Basis:
      Mary – S+have – en+come
   b. **T_not**:
      Mary – S+have+not – en+come
   c. **Affix Hopping**
      Mary – have+S+not – en+come
   d. **Do Support**:
      – (Kann nicht applizieren, weil es kein allein stehendes Affix gibt).
   e. **Realisierung**:
      Mary has not come.

(18) ‘Do’-Einsetzung bei Negation ohne Auxiliar:
   a. Basis:
      Mary – S – come
   b. **T_not**:
      Mary – S+not – come
   c. **Affix Hopping**:
      Mary – have+S+not – en+come
   d. **Do Support**:
      Mary – do+S+not come
   e. **Realisierung**:
      Mary did not come.

Rekonstruktion bei Speas (1995)

(19) Beschränkungsordnung:
ECP ⇒ LETZT-AUS ⇒ ÖKON
$T_{11}$: Negation und ‘do’-Einsetzung bei Speas

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>ECP</th>
<th>LETZT-AUS</th>
<th>ÖKON</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg K_1$: Mary did$_1$ t$_1$ leave</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>$K_2$: Mary t$_1$ not left$_1$</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$T_{12}$: Verbot der ‘do’-Einsetzung ohne Negation bei Speas

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>ECP</th>
<th>LETZT-AUS</th>
<th>ÖKON</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_1$: Mary did$_1$ t$_1$ leave</td>
<td></td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>$\neg K_2$: Mary t$_1$ left$_1$</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Rekonstruktion von Chomsky (1991):

“Um die korrekten Ergebnisse zu erzielen, muss das Prinzip des ‘letzten Auswegs’ so interpretiert werden, dass Prinzipien der UG angewendet werden, wann immer das möglich ist, und dass sprachspezifische Regeln nur benutzt werden, um eine D-Struktur-Repräsentation zu “retten”, die ansonsten kein Ergebnis liefert” (Chomsky (1991, 427)). (Vgl. auch die Logik bei Adgers (2003) Erklärung über die “Pronouncing Tense Rule” (PTR).)

(20) Unabhängig blockiert:
- *Mary left$_1$ not t$_1$
- *Mary left$_1$ t$_1$

Bemerkung:
Die Analyse hat Konsequenzen für die Definition der Kandidatenmenge.

Rekonstruktion bei Grimshaw (1997)

(21) a. **LEX-ÖKON** („Bewegungssökonomie für lexikalische Köpfe“, „No-Lex-Mvt“):
  Bewegung von lexikalischen Köpfen ist verboten (X$^0$-Spur$_{loc}$ ist nicht erlaubt).
  
  b. **KASUS** („Case“; Chomsky (1981)):
  Der Kopf einer NP-Kette muss in einer Kasusposition sein.
  
  c. **OB-KÖPF** („Obligatorische Köpfe“, „Ob-Hd“):
  Eine Projektion hat einen (nicht-leeren) Kopf.
  
  d. **SUBJEKT** („Erweitertes Projektionsprinzip“, „Subj“; Chomsky (1982; 1995)):
  Der höchste A-Spezifikator eines Satzes muss durch ein Argument gefüllt sein.
  
  e. **VOLL-INT** („Vollständige Interpretation“, „Full Interpretation“; Chomsky (1986b)):
  Expletiveinsetzung ist verboten.
  
  f. **ÖKON** („Stay“, s.o.):
  Bewegung ist verboten (Spur ist nicht erlaubt).

(22) **Beschränkungsordnung:**

LEX-ÖKON $\gg$ KASUS $\gg$ OB-KÖPF $\gg$ SUBJEKT $\gg$ VOLL-INT $\gg$ ÖKON

(23) **VOLL-INT:**

Die lexikalisch-konzeptuelle Struktur muss respektiert werden.

**Grundannahme:**

Die Größe der Satzstruktur ist variabel.

(24) a. $\neg$Neg$P$ Not [VP Mary left ]

b. *Neg$P$ Mary$_1$ not [VP t$_1$ left ]

c. *[IP Mary$_1$ [t ] [Neg$P$ not [VP t$_1$ left ]]]

d. *[IP Mary$_1$ [t t$_2$ ] [Neg$P$ not [VP t$_1$ t$_2$ ]]]

e. *[IP Mary$_1$ [t did ] [Neg$P$ not [VP t$_1$ leave ]]]

$T_{13}$: Negation und ‘do’-Einsetzung bei Grimshaw

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>LEX-ÖKON</th>
<th>KASUS</th>
<th>OB-KÖPF</th>
<th>SUBJEKT</th>
<th>VOLL-INT</th>
<th>ÖKON</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_1$: [Neg [VP NP$_1$ V ]]</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>$K_2$: [NP$_1$ Neg [VP t$_1$ V ]]</td>
<td></td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_3$: [NP$_1$ [t Neg [VP t$_1$ V ]]]</td>
<td></td>
<td>*!</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_4$: [NP$_1$ V$_2$ [t Neg [VP t$_1$ t$_2$ ]]]</td>
<td></td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\neg K_5$: [NP$_1$ did$_2$ [t Neg [VP t$_1$ V ]]]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*!</td>
</tr>
</tbody>
</table>

(25) a. [VP Mary left ]

b. *[IP Mary$_1$ [t ] [VP t$_1$ left ]]

c. *[IP Mary$_1$ [t t$_2$ ] [VP t$_1$ t$_2$ ]]

d. *[IP Mary$_1$ [t did ] [VP t$_1$ leave ]]

$T_{14}$: Verbot der ‘do’-Einsetzung ohne Negation bei Grimshaw

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>LEX-ÖKON</th>
<th>KASUS</th>
<th>OB-KÖPF</th>
<th>SUBJEKT</th>
<th>VOLL-INT</th>
<th>ÖKON</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg K_1$: [VP NP$_1$ V ]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_2$: [NP$_1$ [t Neg [VP t$_1$ V ]]]</td>
<td></td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_3$: [NP$_1$ V$_2$ [t Neg [VP t$_1$ t$_2$ ]]]</td>
<td></td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K_4$: [NP$_1$ did$_2$ [VP t$_1$ V ]]]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*!</td>
<td></td>
</tr>
</tbody>
</table>

**Gemeinsamkeit:**

Die Rolle des ECP in Speas’ Analyse und die Rolle von KASUS in Grimshaws Analyse sind äquivalent; es sind die entscheidenden Auslöser von do-Einsetzung in Negationskontexten. Ebenso äquivalent sind LETZT-AUS bei Speas und VOLL-INT bei Grimshaw: Beide Beschränkungen bestrafen die Verwendung eines expletiven Verbs do; sie sind in (Auxiliar-freien) Negationskontexten von optimalen Kandidaten verletzbar. LEX-ÖKON ist bei Speas implizit, bei Grimshaw explizit angenommen; dasselbe gilt auch für SUBJEKT (denn bei Annahme der Hypothese der prädikats-internen Subjekte muss auch Speas...
gewährleisten, dass ein Satz wie "Not Mary left ausgeschlossen ist). Schließlich hat auch ÖKON in beiden Analysen dieselbe Funktion, nämlich interessanterweise – für die betrachteten Fälle – gar keine.

3.2.2 Resumptive Pronouns

(26) Resumptive pronouns in English

a. (the man) who(m) I saw t
b. *(the man) who(m) I don’t believe the claim that anyone saw t
c. *(the man) who(m) I saw him
d. ?(the man) who(m) I don’t believe the claim that anyone saw him

Observation:
The insertion of resumptive pronouns may (often) be viewed as a repair phenomenon, i.e., a last resort, if a well-formed sentence cannot otherwise be generated (Shlonsky (1992), Hornstein (2001)). The insertion of a resumptive pronoun (which, by assumption, is not part of the input) violates a faithfulness constraint, but is required by a higher-ranked markedness constraint. (See Pesetsky (1998), Legendre, Smolensky & Wilson (1998), and Salzmann (2006) for OT analyses of resumptive pronouns.)

(27) a. REL-Criterion (REL-CRIT):
   Relative pronouns are in SpecC of a relative clause.
   b. Complex NP Condition, CNPC):
      A moved item must not be separated from its trace by an intervening DP.
   c. Inclusiveness (Incl, a DEP constraint):
      Every element of the output must be present in the input.

(28) Ranking:
   REL-CRIT $\gg$ CNPC $\gg$ Incl.

$T_{15}$: Trace vs. resumptive pronouns; transparent context

<table>
<thead>
<tr>
<th>Input: I, who(m), saw, C[rel], the, man</th>
<th>REL-CRIT</th>
<th>CNPC</th>
<th>Incl</th>
</tr>
</thead>
<tbody>
<tr>
<td>?? $O_1$: the man who(m) I saw t</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$O_2$: the man who(m) I saw him</td>
<td></td>
<td></td>
<td>$!$</td>
</tr>
<tr>
<td>$O_3$: the man I saw who(m)</td>
<td></td>
<td>$!$</td>
<td></td>
</tr>
</tbody>
</table>

$T_{16}$: Trace vs. resumptive pronoun, opaque CNPC context

<table>
<thead>
<tr>
<th>Input: anyone, who(m), saw, I, do, not, believe, the, claim, that, C[rel], the man</th>
<th>REL-CRIT</th>
<th>CNPC</th>
<th>Incl</th>
</tr>
</thead>
<tbody>
<tr>
<td>$O_1$: the man who(m) I don’t believe the claim that anyone saw t</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>?? $O_2$: the man who(m) I don’t believe the claim that anyone saw him</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$O_3$: the man I don’t believe the claim that anyone saw who(m)</td>
<td></td>
<td></td>
<td>$!$</td>
</tr>
</tbody>
</table>

3.2.3 Other Repair Phenomena

Further cases of repair phenomena:

- Ersatz-infinitive in German, Participium pro Infinitivo in Swedish (Wiklund (2001)) → Schmid (2005)
- R-pronouns in German, Dutch and (Middle) English (Riemsdijk (1978)) → Müller (2000a)
- wh-scope marking in German and Hungarian → Müller (1997)
- repair-driven intermediate movement steps as required by the PIC (Chomsky (2001; 2008)) → Heck & Müller (2000; 2003)

(29) Ersatz-infinitive:

a. dass sie das gehabt *hat
   that she that wanted.PART
b. *dass sie dass haben
   that she that want.INF
   *wollen
   wants
   c. *dass sie das Lied singen gewollt hat
   that she the song sing wanted.PART
   *wollen
   wants

(30) Repair-driven quantifier raising:

a. [CP, Some boy admires every teacher], [ and [CP, some girl does [VP admire every teacher] too ]]

   ($\exists y, \forall z$)

b. [CP, Some boy admires every teacher], [ and [CP, Mary does [VP admire every teacher] too ]]

   ($\exists y, \forall z, *\exists z$)

c. [CP, Mary admires every teacher], [ and [CP, some boy does [VP admire every teacher] too ]]

   ($\exists y, \forall z$)
(31) Repair-driven multiple wh-movement:

a. Irgendjemand hat irgendetwas geerbt, aber der Fritz weiß nicht mehr
someone has something inherited but the Fritz knows not more

\[ \text{CP vor \_ was } \text{gen} \text{ geerbt } \text{but} \]
who what inherited has

b. *Irgendjemand hat irgendetwas geerbt, aber der Fritz weiß nicht mehr
someone has something inherited but the Fritz knows not more

\[ \text{CP vor \_ was } \text{gen} \text{ geerbt } \text{but} \]
who what inherited has

3.3 Default Contexts
Profile of the empirical evidence:
The empirical evidence suggests that there is a concept like “unmarked case” (“default case”, “elsewhere case”): Some linguistic property P of LEs counts as the unmarked case if it shows up whenever something else (that is incompatible with P) is not explicitly required. In standard conceptions of grammar, the theoretical implementation of this concept is far from unproblematic. (Whenever it seems to be unproblematic, as in approaches to syntax that envisage blocking (Williams (1997), Fanselow (1991)), or in Distributed Morphology (Halle & Marantz (1993)), this is due to the fact that the approach is in fact based on competition and candidate sets, too.) In OT, an unmarked case signals the activity of a constraint that is ranked very low, and that is typically rendered inactive by higher-ranked, conflicting constraints → emergence of the unmarked.

Empirical generalization:
In the unmarked case, a DP bears nominative case in German; nominative is the default case.

(32) Nominative as an unmarked case:

a. Constructions with ‘als’ that do not have case agreement:
die Ehreng des Kanzlers als großer Politiker/*großen Politiker
the homage to the chancellor as great politiciannom great politicanacc

b. Infinitival constructions with (case-less) PRO:
Wir baten die Männer [CP PRO einer nach dem anderen]/*einen nach we asked the men nom oneacc after the other oneacc after
dem anderen durch die Sperre zu gehen]
the other through the barricade to go

c. Left dislocation without case agreement:
Der Kaiser/*Den Kaiser, dem verdanken wir nichts
the emperornom the emperornom owe nothing

d. Predicate ‘und’-constructions:
Der/*Den und ein Buch lesen? (Dass ich nicht lache!)
he_nom him_acc and a book read that I not laugh

(33) Blocked nominative in contexts with case government:

a. dass ich *er/ihn getroffen habe
that I he_nom him_acc met have

b. dass man *der Mann/ des Maness gedachte
that one man_nom man_gen remembered

c. dass wir *der Mann/ den Mann das Buch lesen sehen
that we man_nom man_gen the book read see

(34) Case-related constraints:

a. GEN(ITIVE) CONSTRAINT (GEN):
The object of a verb that is lexically marked as governing genitive case bears genitive.
(A subcase of a more general constraint demanding faithfulness to lexical case specifications.)

b. ACC(USATIVE) CONSTRAINT (ACC):
The object of a transitive verb bears accusative case.

c. NOMINATIVE CONSTRAINT (Nom):
A DP bears nominative case.

(35) Ranking:
GEN ≫ ACC ≫ NOM

T_{17}: Accusative government

<table>
<thead>
<tr>
<th>Input: dass, getroffen, habe, 1.Sg./Agent, 3.Sg./Patient</th>
<th>GEN</th>
<th>ACC</th>
<th>Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>**</td>
<td>O₁: dass ich ihm getroffen habe</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>O₂: dass ich er getroffen habe</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₃: dass mich ihm getroffen habe</td>
<td>**</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

T_{18}: Nominative as the unmarked case

<table>
<thead>
<tr>
<th>Input: und, ein, Buch, lesen, 3.Sg./Agent/Dem</th>
<th>GEN</th>
<th>ACC</th>
<th>Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>- O₁: Den und ein Buch lesen?</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>**</td>
<td>O₂: Der und ein Buch lesen?</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O₃: Dem und ein Buch lesen?</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note:
If nominative (absolutive in ergative alignment patterns) is inherently a default case across languages, free reranking in (35) must be blocked in some way (that may be related to the more primitive feature structures of the cases; see, e.g., Wunderlich (1997), Kiparsky (1999; 2001)).

Observation:
Instead of a system of constraints demanding case realization, one can just as well
have a system of constraints prohibiting case realization, accompanied by an inherently highest-ranked (or inviolable) constraint that states that all DPs have case. See Woolford (2001).

**Background assumptions (Woolford (2001))**:

1. There are (ordered) markedness constraints that block the realization of cases.
2. There are faithfulness constraints that demand the realization of case specifications in the input (lexical, inherent case).
3. Nominative/absolutive and accusative are structural cases; dative and ergative (and genitive) are inherent cases (that must be specified on a verb).
4. Every DP must be case-marked (perhaps part of Gen).

(36) a. *Dat ("Dative"):
Avoid dative case.
b. *Acc ("Accusative"):
Avoid accusative case.
c. *Nom ("Nominal"):
Avoid nominative case.
d. FAITH-LEX:
Realize a case feature specified on V in the input.
e. FAITH-LEXtrans:
Realize a case feature specified on transitive V in the input.

(37) a. Ranking in Icelandic:
FAITH-Lextr \( \gg \) FAITH-Lex \( \gg \) *Dat \( \gg \) *Acc \( \gg \) *Nom
b. Ranking in Japanese:
FAITH-Lextr \( \gg \) *Dat \( \gg \) FAITH-Lex \( \gg \) *Acc \( \gg \) *Nom
c. Ranking in English:
*Dat \( \gg \) FAITH-Lextr \( \gg \) FAITH-Lex \( \gg \) *Acc \( \gg \) *Nom

(38) Quirky case in Icelandic:

a. Bátnum hvöldi
boata\_dat capsized
b. Barninu batnadli veikin
child\_dat recovered from disease\_nom

**T19: Intransitive V in Icelandic; inherent dative**

<table>
<thead>
<tr>
<th>Candidates</th>
<th>FAITH-Lextr</th>
<th>FAITH-Lex</th>
<th>*Dat</th>
<th>*Acc</th>
<th>*Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bullet ) O1: DP_dat V_[+dat]</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2: DP_nom V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O3: DP_acc V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**T20: Transitive V in Icelandic; inherent dative on DP\_ext**

<table>
<thead>
<tr>
<th>Candidates</th>
<th>FAITH-Lextr</th>
<th>FAITH-Lex</th>
<th>*Dat</th>
<th>*Acc</th>
<th>*Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bullet ) O1: DP_dat V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2: DP_nom V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O3: DP_acc V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(39) Quirky case in Japanese:

a. Akatyan-ga/*-ni moo arukeru
baby\_nom/dat already walk can
b. Taroo-ni eigo-ga hanaseru
Taro\_dat English\_nom speak can

**T21: Intransitive V in Japanese; no inherent dative**

<table>
<thead>
<tr>
<th>Candidates</th>
<th>FAITH-Lextr</th>
<th>*Dat</th>
<th>FAITH-Lex</th>
<th>*Acc</th>
<th>*Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1: DP_dat V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2: DP_nom V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**T22: Transitive V in Japanese; inherent dative on DP\_ext**

<table>
<thead>
<tr>
<th>Candidates</th>
<th>FAITH-Lextr</th>
<th>*Dat</th>
<th>FAITH-Lex</th>
<th>*Acc</th>
<th>*Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bullet ) O1: DP_dat V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2: DP_nom V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O3: DP_acc V_[+dat]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note**


4. Problems for OT Analyses in Syntax

**Problems for OT Analyses**:

- complexity of competition-based grammars (potentially infinite candidate sets)
- ineffability (absolute ungrammaticality)
- optionality

4.1 Complexity

**Observation**:

Because of the general option of recursion in syntax, candidate sets are not finite in most analyses.
This qualm arises from a misapprehension about the kind of thing that grammars are. It is not incumbent upon a grammar to compute, as Chomsky has emphasized repeatedly over the years. A grammar is a function that assigns structural descriptions to sentences; what matters formally is that the function is well-defined. The requirements of explanatory adequacy (on theories of grammar) and descriptive adequacy (on grammars) constrain and evaluate the space of the hypotheses. Grammatical theorists are free to contemplate any kind of formal device in pursuit of these goals; indeed, they must allow themselves to range freely if there is to be any hope of discovering decent theories. Concomitantly, one is not free to impose arbitrary additional meta-constraints (e.g. “computational plausibility”) which could conflict with the well-defined basic goals of the enterprise. In practice, computationalists have always proved resourceful. All available complexity results for known theories are stunningly distant from human processing capacities ... yet all manner of grammatical theories have nonetheless been successfully implemented in parsers, to some degree or another, with comparable efficiency ... There are neither grounds of principle nor grounds of practicality for assuming that computational complexity considerations, applied directly to grammatical formalisms, will be informative.

Prince & Smolensky (1993, 197; 2004, 233)

Also cf.:

To avoid what has been a continuing misunderstanding, it is perhaps worthwhile to reiterate that a generative grammar is not a model for a speaker or hearer. It attempts to characterize in the most neutral possible terms the knowledge of the language that provides the basis for actual use of language by a speaker-hearer. When we speak of a grammar as generating a sentence with a certain structural description, we mean simply that the grammar assigns this structural description to the sentence. When we say that a sentence has a certain derivation with respect to a particular generative grammar, we say nothing about how the speaker or hearer might proceed, in some practical or efficient way, to construct such a derivation. These questions belong to the theory of language use – the theory of performance.

Chomsky (1965, 9)

Note:
If there is a problem here, OT shares the problem with other competition-based theories of syntax (e.g., early minimalist approaches like that of Chomsky (1993), which rely on transderivational constraints applying to candidate derivations in large (typically infinite) reference sets).

4.2 Ineffability (Absolute Ungrammaticality)

Observation:
Basically, a sentence can only qualify as ungrammatical if there is some other sentence that blocks it by being the optimal candidate. However, sometimes it is far from obvious what this other sentence should look like.

Adjunct islands in German:
*Was ist Fritz eingeschlafen [CP nachdem er t gelesen hat] ?
what is Fritz fallen asleep after he read has

4.2.1 The Generator

Assumption:
Gen contains constraints like (41) that preclude a generation of outputs in the first place.

(41) Adjunct Condition:
Movement must not cross an adjunct clause.

4.2.2 Empty Outputs

Assumption:
Each candidate set contains a candidate that leaves the input completely unrealized. This candidate is the “empty output” or “null parse”: Ø. By definition, the empty output does not violate any faithfulness constraints; the only constraint that it violates is (42).

(42) *Ø (“Avoid Null Parse”):
The input must not be completely unrealized.

T23: Ineffability and empty outputs

<table>
<thead>
<tr>
<th></th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>Ø</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1: was ... [nachdem er t V]</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2: – ... [nachdem er was V]</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ø = Ø, Ø</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Note:
The constraint *Ø defines a strict upper bound in constraint rankings: Constraints that outrank *Ø are not violable by optimal outputs.

4.2.3 Bad Winners

Assumption:
The optimal candidate cannot be interpreted by other components of grammar (phonology, semantics), or by the interfaces with these components.

T24: Ineffability and bad winners

<table>
<thead>
<tr>
<th></th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1: was ... [nachdem er t V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>Ø = Ø, Ø</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note:
This approach arguably chimes in well with recent trends in minimalist syntax to attribute much of the work standardly done by syntactic constraints to interface requirements; see Chomsky (2007; 2008), and particularly Boeckx (2009).
4.2.4 Repair

Hypothesis:
There is in fact an optimal repair candidate for (40).

(44) Two potential repair candidates:
  a. Fritz ist eingeschlafen [CP nachdem er was gelesen hat] (= etwas)
      Fritz is fallen asleep after he something read has
  b. Bei was ist Fritz eingeschlafen [CP nachdem er es gelesen hat] ?
      with respect to what is Fritz fallen asleep after he it read has

(45) MAX([wh]):
A feature [+wh] in the input must be realized in the output.

T25: Ineffability and repair

<table>
<thead>
<tr>
<th></th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1:</td>
<td>was_{[+wh]} ... [nachdem er t V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O2:</td>
<td>– ... nachdem er was_{[+wh]} V</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>*# O3:</td>
<td>– ... nachdem er was_{[wh]} V</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(46) Problem:
  a. Long wh-movement:
     Was glaubt Fritz [CP dass er t lesen sollte] ?
     what thinks Fritz that he read should
  b. Wh-indefinite:
     Fritz glaubt [CP dass er was lesen sollte] 
     Fritz thinks that he what (= something) read should
  c. Optional wh-argument plus resumptive pronoun:
     Von was glaubt Fritz [CP dass er es lesen sollte] ?
     of what thinks Fritz that he it read should

T26: A wrong prediction

<table>
<thead>
<tr>
<th></th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>*# O1:</td>
<td>was_{[+wh]} ... [dass er t V]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2:</td>
<td>– ... [dass er was_{[+wh]} V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O3:</td>
<td>– ... [dass er was_{[wh]} V]</td>
<td>*!</td>
<td></td>
</tr>
</tbody>
</table>

Conclusion:
Wh-indefinite clauses are not repair forms; they are available even if long wh-movement is permitted. (Similar conclusions hold in the case of sentences with optional wh-argument generation in the matrix clauses; see Koster (1986), Cinque (1990), Barbiers (2002), Gal-

4.2.5 Neutralization

Assumption:
There are two competitions based on minimally differing inputs (e.g., inputs that differ only with respect to some feature value). These input differences can then be neutralized by some high-ranked markedness constraint in the output; i.e., two different competitions (based on two candidate sets) converge on a single optimal candidate.

T27: Transparent contexts without neutralization: ‘was_{[+wh]}’ in the input

<table>
<thead>
<tr>
<th>Input:</th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>*# O1:</td>
<td>was_{[+wh]} ... [dass er t V]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2:</td>
<td>– ... [dass er was_{[+wh]} V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O3:</td>
<td>– ... [dass er was_{[wh]} V]</td>
<td>*!</td>
<td></td>
</tr>
</tbody>
</table>

T28: Transparent contexts without neutralization: ‘was_{[wh]}’ in the input

<table>
<thead>
<tr>
<th>Input:</th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>*# O1:</td>
<td>was_{[wh]} ... [dass er t V]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2:</td>
<td>– ... [dass er was_{[wh]} V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O3:</td>
<td>– ... [dass er was_{[wh]} V]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note:
A further output O4 that applies movement of a [-wh] phrase to SpecC in T28 has the same constraint profile as O3 with respect to the three constraints given here. However, it is suboptimal (in fact, harmonically bounded, i.e., not expected as grammatical under any ranking) because it violates economy in addition without contributing to a better behaviour with respect to any other constraint.

(47) Competing outputs:
  a. *Was ist Fritz eingeschlafen [CP nachdem er t gelesen hat] ?
     what is Fritz fallen asleep after he read has
  b. Fritz ist eingeschlafen [CP nachdem er gelesen hat]
     Fritz is fallen asleep after he what (= something) read has

T29: Island contexts with neutralization, unfaithful: ‘was_{[+wh]}’ in the input

<table>
<thead>
<tr>
<th>Input:</th>
<th>Adjunct Condition</th>
<th>Wh-Crit</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1:</td>
<td>was_{[+wh]} ... [nachdem er t V]</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>O2:</td>
<td>– ... nachdem er was_{[+wh]} V</td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>*# O3:</td>
<td>– ... nachdem er was_{[wh]} V</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legends:
- CP: Cleft Passive
- O: Output
- Wh: wh
- Adjunct: Adjunct
- Wh-Crit: Wh-Critical
- Adjunct Condition: Adjunct Condition
- Wh-Crit: Wh-Critical
- MAX([wh]): MAX([wh])
T30: Island contexts with neutralization, faithful: ‘was[−wh]’ in the input

<table>
<thead>
<tr>
<th>Input: was[−wh], ...</th>
<th>ADJUNCT</th>
<th>WH-CRIT</th>
<th>MAX([wh])</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1: was[−wh], ... [ nachdem er t V]</td>
<td>![ ]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
<tr>
<td>O2: ... [ nachdem er was[−wh] V]</td>
<td>![ ]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
<tr>
<td>O3: ... [ nachdem er was[−wh] V]</td>
<td>![ ]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
</tbody>
</table>

Note:
For more on ineffability, see Fanselow & Féry (2002a), Legendre (2009), Vogel (2009).

4.3 Optionality

Observation:
In general, only one candidate should be optimal. (However, note that the definition of optimality in (4) is in principle compatible with there being more than one winner, in contrast to, e.g., the one in Grimshaw (1997).) What about situations where it looks as though several outputs can co-exist as optimal?

(48) Complementizer deletion in English:
   a. I think – John will leave
   b. I think that John will leave

(49) Partial wh-movement in German:
   a. Wen glaubst du [CP dass man t einladen sollte ] ?
   b. Was glaubst du [CP wen man t einladen sollte ] ?

(50) Wh-movement in French:
   a. Qui as-tu vu t ?
   b. – Tu as vu qui ?

(51) Extraposition in German:
   a. dass eine Frau [ die ich mag ] zur Tür reingekommen ist
   b. dass eine Frau t zur Tür reingekommen ist [ die ich mag ]
   that a woman whom I like to the door in come is
   that a woman to the door in come is whom I like

(52) Free word order in German:
   a. dass keiner den Fritz gesehen hat
   b. dass den Fritz keiner gesehen hat
   that no-one the Fritz acc seen has
   that the Fritz acc no-one theacc seen has

4.3.1 Pseudo-Optionality

Assumption:
Candidate sets are defined in such a way that there is little competition.

Problem:
If there is not much competition, this weakens the overall theory and increases the problem of accounting for ineffability.

(54) Competition of partial and long-distance wh-movement in German:
   a. ?Wen glaubst du nicht [CP dass man t einladen sollte ] ?
   whom think you not that one invite should
   b. *Was glaubst du nicht [CP wen man t einladen sollte ] ?
   what think you whom one invite should

(55) Competition of wh-movement and wh-in-situ in French:
   a. Je me demande [ qui C tu as vu t ]
   I ask myself whom dyou have seen
   b. *Je me demande [ – (que) tu as vu qui ]
   I ask myself that you have seen whom

4.3.2 True Optionality

Assumption:
Two (or more) candidates can in fact have the same (optimal) constraint profile.

Problem:
This proves very hard (or impossible) in practice, e.g., because of the existence of faithfulness constraints.
4.3.3 Ties

Assumption:
Two (or more) constraints are equally important, i.e., tied. Candidates that differ only with respect to these ties can all be optimal.

\[ T_{31}: \text{Constraint tie: } B \triangleright C \]

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\triangleright) O₁</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>(\triangleright) O₂</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₃</td>
<td>*(!)</td>
<td>*(!)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O₄</td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note:
There are various different (and largely incompatible) concepts of tie in the literature. A basic distinction that can be made is one between global ties and local ties. Global ties are abbreviations for the simultaneous presence of different constraint rankings in a language. Local ties can be viewed as special constraint types. The most widespread concepts of tie in the literature are arguably ordered global ties and conjunctive local ties.

Observation:
With global ties, the optimal outputs of one candidate set may have a different constraint profile below the tie. With local ties, such a different behaviour with lower-ranked constraints leads to a breakdown of optionality.

(56) Ordered global tie:
Suppose that \( \Gamma = < \text{Con}_1 \triangleright \ldots \triangleright \text{Con}_n > \) is a partial constraint order in language L, and \( \text{Con}_i \) (\( 1 \leq i \leq n \)) is a tie \( \text{Con}_{i_1} \circ \ldots \circ \text{Con}_{i_m} \). Then, for every suborder \( O \) of the constraints in \( \text{Con}_i \), \( \Gamma_O \) is a constraint order of language L, where \( \Gamma_O \) differs from \( \Gamma \) only in that \( \text{Con}_i \) is replaced by \( O \).

(57) Diagram of an ordered global tie B\( \triangleright \)C

\[
A \triangleright B \Rightarrow D
\]

\[
C \triangleright B \Rightarrow D
\]

4.3.4 Stochastic Optimality Theory

Refs.:
Aissen (2002; 2003), Bresnan, Dingare & Manning (2001), Bresnan, Deo & Sharma (2007) (syntactic applications)

Observation:
Often, the constructions that participate in an alternation are not equally frequent or equally unmarked (or, for that matter, equally “well formed” – degrees of acceptability).

(60) Preferences with optionality in the positioning of English possessives:
\begin{itemize}
  \item a. the result of the accident \( > \) the accident’s result
  \item b. Mary’s sister \( > \) the sister of Mary
  \item c. the boy’s uncle \( > \) the uncle of the boy
  \item d. the door of the building \( > \) the building’s door
  \item e. someone’s shadow \( > \) the shadow of someone
  \item f. the shadow of something \( > \) *something’s shadow
  \item g. her money \( ? > \) *the money of her
\end{itemize}

Observation:
Animacy and definiteness scales are independently motivated (Hale (1972), Silverstein (1976)). These hierarchies can be used as primitives to generate sequences of constraints (with a fixed internal order: subhierarchies), via harmonic alignment of scales (Prince & Smolensky (2004), Aissen (1999)).

(61) Harmonic Alignment (Prince & Smolensky (2004)):
Suppose given a binary dimension \( D_1 \) with a scale \( X > Y \) on its elements \( \{X,Y\} \), and another dimension \( D_2 \) with a scale \( a > b > \ldots > z \) on its elements \( \{a,b,\ldots,z\} \). The harmonic alignment of \( D_1 \) and \( D_2 \) is the pair of Harmony scales \( H_X, H_Y \):
\begin{itemize}
  \item a. \( H_X: X/a > X/b > \ldots > X/z \)
  \item b. \( H_Y: Y/z > \ldots > Y/b > Y/a \)
\end{itemize}
The constraint alignment is the pair of constraint hierarchies \( C_X, C_Y \):
\begin{itemize}
  \item a. \( C_X: *X/z > \ldots > *X/b > *X/a \)
  \item b. \( C_Y: *Y/a > *Y/b > \ldots > *Y/z \)
\end{itemize}

(62) Constraint subhierarchies via animacy and definiteness scales:
\begin{itemize}
  \item (i) \( *\text{SpecN}/\text{inanimate} \gg *\text{SpecN}/\text{animate} \gg *\text{SpecN}/\text{human} \)
  \item (ii) \( *\text{CompN}/\text{human} \gg *\text{CompN}/\text{animate} \gg \text{CompN}/\text{inanimate} \)
b. (i) **SpecN/indef \gg SpecN/def \gg SpecN/name \gg SpecN/pron**
   (ii) **CompN/pron \gg CompN/name \gg CompN/def \gg CompN/indef**

Proposal: Constraints are not necessarily categorically ordered with respect to each other. Rather, their application domains may overlap. An overlap of application domains gives rise to optionality.

5. Optimization Domains

5.1 Introduction

Question:
Does syntactic optimization apply once (harmonic parallelism: representational syntax) or more than once (harmonic serialism: derivational syntax)? If the latter holds: Is optimization global or local?

But wait:
Isn’t optimality theory inherently representational/non-derivational?

Answer:
No, it isn’t.

Much of the analysis given in this book will be in the parallel mode, and some of the results will absolutely require it. But it is important to keep in mind that the serial/parallel distinction pertains to Gen and not to the issue of harmonic evaluation *per se*. It is an empirical question [...] Many different theories [...] can be equally well accommodated in Gen, and the framework of Optimality Theory *per se* involves no commitment to any set of such assumptions.

Prince & Smolensky (2004, 95-96)

While some see a major divide between the derivationally-oriented MP and OT, we do not. Of course, there are likely to be differences of empirical import between the non-derivational, chain-based theory of “Shortest Move” developed here and a particular derivational MP proposal, but such differences seem comparable to those between different approaches to syntax within OT, or to those between different proposals within MP: they do not seem to follow from some major divide between the OT and MP frameworks. In fact, derivational theories can be naturally formalized within OT. “Harmonic serialism” is a derivational version of OT developed in Prince & Smolensky (1993) in which each step of the derivation produces the optimal next representation. Another approach, seemingly needed to formalize MP within OT has *Gen* produce derivations; it is these that are evaluated by the constraints, the optimal derivation being determined via standard OT evaluation. Thus, on our view, while the issue of derivations is an important one, it is largely orthogonal to OT.

c. derivational step (multiple optimization, derivational)

**Background:**

(i) Classical assumption: The whole sentence is subject to a single, parallel optimization procedure (Grimshaw (1997), Pesetsky (1998), Legendre, Smolensky & Wilson (1998) etc.). The output candidates are usually taken to be representations; but they can also be full derivations (as, e.g., in Müller (1997)).


(iii) Multiple optimization of smaller optimization domains: closely related to developments in the minimalist program.

**Observation:**

Small optimization domains presuppose a derivational approach to syntax.

**Conceptual argument for small optimization domains:**

The smaller the optimization domain is, the more the complexity of the overall system is reduced (reduction of the size of candidate sets).

**Conceptual argument for larger optimization domains:**

The larger the optimization domain is, the less often optimization procedures have to be carried out.

**Empirical arguments for smaller/larger optimization domains:**

If the ranked constraints have access to more/less structure, a wrong winner is predicted.

(68) **Proposals for local optimization:**

a. **Minimal clause:**

Ackema & Neeleman (1998) on wh-movement in Czech; Müller (2003a) on extraction from verb-second clauses in German

b. **Phase:**

Fanselow & ‘Cavar (2001) on McN-deletion in Malay; Müller (2000a; 2002) on R-pronouns in German

c. **Phrase:**


d. **Derivational step:**


5.2 **Clauses as optimization domains**

Ref.: Ackema & Neeleman (1998)

(69) **Long multiple wh-movement in Czech proceeds without wh-cluster formation:**

\[
\text{[VP} \text{Co1 [VP podle tebe [VP Petr rieł [CP že Jan dal t1 what according to you whom Petr said that Jan gave t2]]]]}
\]

**The proposal:**

Evaluation of movement constraints proceeds cyclically. That is to say, \textit{Stay} is first evaluated with respect to the embedded clause, then to the combination of the embedded clause and the matrix clause. In the embedded clause, \textit{Stay} favours separate movement of the two wh-expressions [...] This means that clustering can only take place when the larger cycle is taken into account, i.e., when the two whs have already been adjoined to the embedded VP. However, it is no longer possible then, because it would have to take place within the embedded clause (the initial landing site of the whs), which would go against strict cyclicity.

Ackema & Neeleman (1998, fn. 25)

(70) a. **Q-Mark:**

Assign [+Q] to a propositional constituent.

(This can only be done by an overt functional head, which in turn needs to inherit this capacity in the matrix clause from some wh-phrase in its specifier.)

b. **Q-Scope:**

[+Q]-elements must c-command the constituent representing the proposition.

c. **Stay:**

Every node crossed by movement induces a violation.

**T32: Long multiple wh-movement in Czech, optimization of embedded CP**

<table>
<thead>
<tr>
<th>Input: part of the numeration</th>
<th>Q-Scope</th>
<th>Stay</th>
<th>Q-Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>O1:</strong> [cp že [vp co1 [vp komu2 [vp Jan dal t1 t2]]]]</td>
<td>***</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>O2:</strong> [cp komu2 že [vp co1 [vp Jan dal t1 t2]]]</td>
<td>***†</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>O3:</strong> [cp co1 komu2 že [vp Jan dal t1 t2]]</td>
<td><em><strong>†</strong></em></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>O4:</strong> [cp že [vp Jan dal co1 komu2]]</td>
<td>†</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notation:**

O11–O13 in \( T_{32} \) are descendants of \( O_1 \).
T₃₃: Long multiple wh-movement in Czech, optimization of matrix clause

<table>
<thead>
<tr>
<th>Input: [cp že [vp co₁ [vp komu₂ [vp Jan dal t₁ t₂]]]] Petr, řekl</th>
<th>Q-Scope</th>
<th>Stay</th>
<th>Q-Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>O₁₁:</strong> [cp co₁ [vp komu₂ [vp Petr řekl [cp že [vp t₁ [vp t₂ [vp Jan dal t₁ t₂]]]]]]]</td>
<td>++++</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td><strong>O₁₂:</strong> [cp co₁ řekl [vp komu₂ [vp Petr [cp že [vp t₁ [vp t₂ [vp Jan dal t₁ t₂]]]]]]]</td>
<td>+++</td>
<td>!+</td>
<td></td>
</tr>
<tr>
<td><strong>O₁₃:</strong> [cp co₁ komu₂ řekl [vp [t₁ t₂] Petr [cp že [vp t₁ [vp t₂ [vp Jan dal t₁ t₂]]]]]]</td>
<td>++++</td>
<td>!+</td>
<td></td>
</tr>
</tbody>
</table>

Note:
Global optimization of the whole sentence would predict a wrong winner: “It seems to be predicted that when the distance to be covered by the wh-expressions in a multiple question increases, clustering [as in Bulgarian, with a high-ranked Q-Mark] will be favoured.”

T₃₄: Global optimization: Long multiple wh-movement in Czech, wrong winner

<table>
<thead>
<tr>
<th>Input: numeration</th>
<th>Q-Scope</th>
<th>Stay</th>
<th>Q-Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>O₁:</strong> [vp co₁ [vp komu₂ [vp Petr řekl [cp že [vp Jan dal t₁ t₂]]]]]</td>
<td>++++!</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td><strong>O₂:</strong> [cp co₁ komu₂ řekl [vp Petr [cp že [vp Jan dal t₁ t₂]]]]</td>
<td>++++</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Underlying logic:
(i) Two short movements are better than a short movement and a longer movement: 2+2=4, 1+5 =6.
(ii) Two medium-sized movements can be worse than a short movement and a very long movement: 7+7=14, 1+10 =11.

5.3 Derivational Steps as Optimization Domains

Premise:

Two constitutive properties:
(i) MP: Syntactic structure is built up derivationally.
(ii) OT: Well-formedness of syntactic objects is determined via optimization.

Combining the properties:
(i) Syntactic structure is built up derivationally and is subject to repeated local optimization: structure building operations and optimization apply in a cyclic interleaving fashion.
(ii) Based on a given input, the operations Merge, Move, and Agree create various output candidates \(\alpha₁, ..., \alpha_n\); the candidate set \(M\) is subject to optimization.
(iii) The optimal output \(\alpha_i\) serves as the input for the next cycle, and so on, until the numeration is empty.
(Note: In assuming structure-building and optimization to systematically alternate, this approach crucially differs from Broekhuis' Derivations and Evaluations model, where all structure-building precedes optimization.)

Aim:
Pursue the consequences of the most radical position within a theory of local optimization: extremely local optimization.

This is tantamount to the claim that each transformational rule application constitutes a “phase,” which we believe to be the null hypothesis.

Epstein & Seely (2002, 77)

Shape of the Argument:
(i) Sometimes, the order of applying Agree and Merge is under-determined. If there are no simultaneous rule applications in the grammar (see Epstein & Seely (2002); contra Pullum (1979), Chomsky (2008)), then a conflict arises: Only one of them can be executed at each step.
(ii) The conflict can be resolved by ranking the requirements: The highest-ranked requirement is satisfied immediately; lower-ranked ones must remain unsatisfied at the current derivational step. Such unsatisfiability does not lead to a crash of the derivation and thus suggests an analysis in terms of violates constraints.
(iii) If the optimization domain is larger than the step-level, then, ceteris paribus, the order of elementary operations that is imposed by the ranking under step-level optimization cannot be preserved. Empirically, this is the wrong result.

5.3.1 Constraints, Features, and Operations
(71) Two types of features that drive operations (see Sternefeld (2003), Adger (2003)):
   a. Structure-building features (edge features, subcategorization features) trigger Merge: [\(\ast F\ast\)].
   b. Probe features trigger Agree: [\(\ast F\ast\)].

(72) Merge:
   \(\alpha\) can be merged with \(\beta\), yielding \(\{\alpha, \{\alpha, \beta\}\}\), if \(\alpha\) bears a structure-building feature [\(\ast F\ast\)] and \(F\) is the label of \(\beta\).

(73) Move:
   Move is Merge, with \(\beta\) internal to \(\alpha\).
(74) **Agree:**
α can agree with β with respect to a feature bundle Γ iff (a) and (b) hold:

a. α bears a probe feature [SF] in Γ and may thereby provide the α-value for a matching goal feature [F] of β in Γ.
b. α m-commands β.

(This permits an Agree relation between a head and its specifier.)

(75) **AGREE CONDITION (AC):**
Probes ([SF]) participate in Agree.

(76) **MERGE CONDITION (MC):**
Structure-building features ([SF]) participate in Merge.

(77) **STRUCT CYCLE CONDITION (SCC, Chomsky (1973; 1993)):**
Merge of α and β is possible only if β has no active features. (A feature is active if it is a [F] or [SF] feature that has not yet participated in Merge or Agree).

(78) **LAST RESORT (LR):**
Move of α and β follows Agree of α and β.

Comment:
The (perhaps less ordinary) treatment of Move in (78) as a binary operation rests on the assumption that Move is Merge (with β internal to α), which is binary.

### 5.3.2 Empirical Evidence for Extremely Local Optimization 1: Argument Encoding

(79) **Basic patterns of argument encoding:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Nom</th>
<th>Acc</th>
<th>Int</th>
<th>Erg</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP_{ext-V}</td>
<td>DP_{int-V}</td>
<td>DP_{ext-V}</td>
<td>DP_{int-V}</td>
<td></td>
</tr>
<tr>
<td>nom</td>
<td>acc</td>
<td>erg</td>
<td>abs</td>
<td></td>
</tr>
</tbody>
</table>

**Note:**
DP_{ext} = external argument DP; DP_{int} = internal argument DP.
V_{i} = intransitive verb; V_{t} = transitive verb.

**Assumptions about argument encoding:**

(i) There is one structural argument encoding feature: [case].
(ii) [case] can have two values: ext(ernal) and int(ernal) (determined with respect to vP).
(iii) [case:ext] = nominative/absolutive, [case:int] = accusative/ergative (see Murasugi (1992)).

(iv) [case] features figure in Agree relations involving T/v and DP, as in (80).

(80) **The role of T and v in argument encoding:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Nom</th>
<th>Acc</th>
<th>Int</th>
<th>Erg</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP_{int-V}</td>
<td>DP_{ext-V}</td>
<td>DP_{int-V}</td>
<td>DP_{ext-V}</td>
<td></td>
</tr>
<tr>
<td>nom</td>
<td>acc</td>
<td>erg</td>
<td>abs</td>
<td></td>
</tr>
</tbody>
</table>

A conspicuous property:
The head v has a dual role: It participates in a Merge operation with a DP, and it also participates in an Agree relation with a DP. This dual role has far-reaching consequences for the nature of argument encoding.

**A constraint conflict:**
Consider a simple transitive context, with two arguments DP_{int}, DP_{ext}. Suppose that the derivation has reached a stage Σ where v has been merged with a VP containing DP_{int}, with DP_{ext} waiting to be merged with v in the workspace of the derivation. At this point, a conflict arises: AC demands that the next operation is Agree(v,DP_{int}) (see (i)), MC demands that it is Merge(DP_{ext},v) (see (ii)). (Application of these constraints at each derivational step derives the effects of Pesetsky’s (1989) Earliness Principle, see Chomsky (2001, 15).)

**Convention:**
A feature [F] whose value is not yet determined is written as “[F,□].”

(81) **Stage Σ:**

```
DP_{case,□}  \[ \rightarrow \]
V_{\text{[case:int,\text{s}]}}[\text{[ext,\text{v}]}] \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \ →
Path (Müller (1998, 130); also cf. Pesetsky (1982, 280), Collins (1994, 56)): The path from X to Y is the set of categories Z such that (a) and (b) hold:

a. Z is reflexively dominated by the minimal XP that dominates both X and Y.
b. Z dominates X or Y.

The length of a path is determined by its cardinality.

Consequences:
(i) The specifier and the complement of a head qualify as equally close to the head.
(ii) The specifier of a head is closer to the head than a category that is further embedded in the complement of the head.
(iii) DP_ext is now closer to v than DP_int.

Proposal:
This conflict of AC and MC is resolved by language-specific constraint ranking; the two possibilities yield accusative and ergative patterns of argument encoding.

(85) Rankings:
   a. Accusative patterns:
      (MLC ≫) AC ≫ MC
   b. Ergative patterns:
      (MLC ≫) MC ≫ AC

T35: Accusative pattern, step 1 (Σ as input): Agree

```
Input: [v' Vcase[int] ... DPcase[int] ... ]
Workspace = {DPcase[int], TPcase[ext] ... }                  MLC AC MC
O1: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
O2: [v' Vcase[int] ... DPcase[int] ... ]
```

T36: Accusative pattern, step 2: Merge

```
Input: [v' Vcase[int] ... DPcase[int] ... ]
Workspace = {DPcase[int], TPcase[ext] ... }                  MLC AC MC
O21: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
```

T37: Accusative pattern, step 3: Merge

```
Input: TPcase[ext] [V' Vcase] ... DPcase[int] [v' V ... DPcase[int] ... ]
Workspace = {TPcase[ext]}                                    MLC AC MC
O21: [TPcase[ext] [V' Vcase] ... DPcase[int] [v' V ... DPcase[int] ... ]]
```

Note:
Agree in T42 is just local enough to be in accordance with the PIC in Chomsky (2001).

T38: Accusative pattern, step 4: Agree

```
Input: [TPcase[ext] [V' Vcase] ... DPcase[int] [v' V ... DPcase[int] ... ]]
Workspace = {TPcase[ext]}                                    MLC AC MC
O211: [TPcase[ext] [V' Vcase] ... DPcase[int] [v' V ... DPcase[int] ... ]]
```

T39: Ergative pattern, step 1 (Σ as input): Merge

```
Input: [v' Vcase[int] ... DPcase[int] ... ]
Workspace = {TPcase[ext], TPcase[ext], ... }                  MLC AC MC
O1: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
O2: [v' Vcase[int] ... DPcase[int] ... ]
```

T40: Ergative pattern, step 2: Agree (with DP_ext)

```
Input: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
Workspace = {TPcase[ext]}                                    MLC AC MC
O11: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
O12: [v' DPcase[int] [v' Vcase[int] ... DPcase[int] ... ]]
```

14). (Also, DP_ext does not intervene, given the definition of the MLC).

(86) VP-internal nominative arguments in Icelandic:
Honum lik-a själ-f-eclk-ir leikar-ar
PRON.3SG.MASC.DAT like-3.PL self-GEN-love-3.PL.NOM actor-PLNOM
'He likes selfish actors.'
(Sigurðsson (2002, 702))

(87) a. Agree before Merge: accusative
b. Merge before Agree: ergative

Less local optimization:
Suppose that the optimization domain is the phrase, the phase, the clause, or the whole sentence. Other things being equal, this makes wrong empirical predictions (a candidate chosen wrongly as optimal is marked ♦, as before): Given the MLC, [c:case:int] can
### T43: vP optimization under MC ⊃ AC (‘ergative’) ranking: right result

<table>
<thead>
<tr>
<th>Input: DC[case:int], V[case:int], Xs; [VP ... DP[case:int] ... ]</th>
<th>Workspace = { }</th>
<th>MC</th>
<th>AC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

never be instantiated on DP\textsubscript{int}, but must be instantiated on DP\textsubscript{ext}: Once DP\textsubscript{ext} is part of the structure, Agree(v, DP\textsubscript{int}) w.r.t. [case] is impossible.

#### 5.3.3 Prenominal Dative Possessors in German

**Observation:**
(i) German exhibits a construction with a dative-marked possessor DP\textsubscript{2} in SpecD of a matrix DP\textsubscript{1} (see, e.g., Haider (1988), Zifonun (2004)).
(ii) D\textsubscript{1} is realized by a possessive pronoun.
(iii) The root of the pronoun agrees with DP\textsubscript{dat} with respect to [num] and [gen].
(iv) The inflection of the pronoun agrees with its complement NP with respect to [num], [gen], and [case]. We focus here on agreement with respect to [gen] (see (88)), but everything can be transferred to the other features as well.

#### (88) Gender agreement with dative possessors in German:

(a) \( [\text{DP dem Fritz }] \text{ sein} \quad \text{e Schwester} \)
the.masc Fritz his.masc -fem sister.fem
“Fritz’s sister”

(b) \( *[\text{DP dem Fritz }] \text{ ihr} \quad \text{-O Schwester} \)
the.masc Fritz her.masc -masc sister.fem
“Fritz’s sister”

**Analysis:**
(i) DP\textsubscript{dat} is merged as a complement of the possessee (de Vries (2005)) and undergoes [EPP\textsuperscript{*}]-driven movement to SpecD.
(ii) Functional elements like pronouns are realized by post-syntactic morphology (see, e.g., Halle & Marantz 1993).
(iii) The pronoun’s inflectional features occupy a structurally higher position than its root (√ features).
Less local optimization:
Suppose optimization applied to phrases. An optimal DP will always involve raising of DPdat. But with DPdat raised, both DPdat and NP are equally close to the pronoun. Then the inflectional probe can receive value [masc], deriving (88-b) (see O2 in T47): Thus the approach overgenerates.

5.3.4 Conclusion

(90) Results:

a. Extremely local optimization in syntax seems viable.
b. Extremely local optimization in syntax is supported empirically:
   (i) There are indeterminacies in rule application (Agree vs. Merge) in the minimalist program that need to be resolved.
   (ii) They can be resolved in a principled way by assuming constraint violability and constraint ranking, i.e., standard optimality theory (the harmonic serialism version of Prince & Smolensky (2004)).
   (iii) The evidence from argument encoding patterns and prenominal dative possessors suggests that optimization is extremely local, affecting the single operation: Less local optimization produces wrong results because differences that can be detected in the derivation may be lost at the phrase (hence: phrase, clause, sentence) level.

Outlook:
There are certain repair phenomena that lend themselves to an optimality-theoretic solution but initially raise problems in a local optimization approach because they involve long dependencies.

(91) Optimization procedures that are apparently non-local:

a. Resumptive pronouns (as in (26), see Legendre, Smolensky & Wilson (1998), Pesetsky (1998))
b. Long-distance binding (see Fischer (2004))

Hypothesis:
These phenomena all involve issues of morphological realization in chains. In these cases (and only in these), optimization can be non-local: Chains provide wormholes for post-syntactic spell-out (Fischer (2004; 2006)).

6. Ubiquity of Optimization

6.1 Background

State of affairs:
As a common research program, optimality-theoretic syntax is not well.
(i) There are no regular workshops expressly devoted to optimality-theoretic syntax anymore. (There were such workshops on optimality-theoretic syntax between 1997 and 2002, originally initiated by Sten Vikner at Stuttgart University, and there were several such meetings in the US in the second half of the last decade.)
(ii) Very few optimality-theoretic syntax papers have appeared in leading journals over the last few years.
(iii) The few papers that have appeared in the last years do not seem to share common research goals, do not tackle similar questions, and regularly do not cite other recent work in optimality-theoretic syntax.
(iv) For some time, new edited volumes with a focus on optimality-theoretic syntax appeared regularly (see, e.g., Archangeli & Langendoen (1997), Dekkers et al. (2001), Fanselow & Féry (2002b), Legendre, Grimshaw & Vikner (1998), Müller & Sternefeld (2001), Sells (2001b)). This seems to have stopped. (Note also that the working paper volumes Vogel & Broekhuis (2006) and Broekhuis & Vogel (2008) on “Optimality Theory and Minimalism” both have only few contributions that might rightfully be subsumed under the label “optimality-theoretic syntax”. And the book series Advances in Optimality Theory, edited by Ellen Woolford and Armin Mester (Equinox publishing), does not seem to have a single volume yet that would be (mainly) on syntax, let alone a syntax monograph.)
(v) All this is very different from the situation in phonology, morphology, semantics, and pragmatics (see particularly the work on bidirectional OT going back to Blutner (2000) and Jäger & Blutner (2000), and thriving to this day). That said:

The legacy of optimality-theoretic syntax:
Optimality-theoretic syntax lives on. Its key mechanisms are implicit in much recent (and not so recent) work in the Principles and Parameters tradition, and optimization procedures arguably form an indispensable part of the minimalist program.

6.2 Hidden Optimization

Observation:
Implicit optimization procedures (that must be construed with violable and ranked constraints) show up again and again in work in the Principles and Parameters tradition.
Some instances of hidden optimization:

- **OblControl** $\gg$ *Pron
  analysis of pronoun vs. PRO in English gerunds via Avoid Pronoun in Chomsky (1981); see Müller (2000b)
- **Top/PRO** $\gg$ *Pron
  analysis of pro vs. overt pronoun in pro-drop languages via Avoid Pronoun in Haege-eman (1995); see Müller (2000b)
- **Faith(Lex)** $\gg$ OblControl
  analysis of lexical vs. structural control in German in von Stechow & Sternefeld (1988); see Müller (2000b)
- **IP-Crit** $\gg$ Wh-Crit
  analysis of complementizer-finality and the absence overt wh-movement in Japanese in Kayne (1994); see Müller (2000b)
- *Complex-WH** $\gg$ Wh-Real
  analysis of multiple wh-questions in German in Grewendorf (2001); see Müller (2000b)
- *ComplexHead** $\gg$ Head-Real
  phonological realization in head chains in Roberts (1997); see Müller (2000b)
- **N-Def**, **D-Def**, **HMC** $\gg$ **N-to-D** $\gg$ *Dissoc, Full-Int** (Swedish)
  D-Def, HMC $\gg$ N-to-D $\gg$ *Dissoc, Full-Int, N-Def** (Danish)
  analysis of definiteness marking in Swedish and Danish DPs in Embick & Noyer (2001); see Heck & Müller & Trommer (2008)
- **Minimality, Feature Matching** $\gg$ Agree
  analysis of Agree in Haege-eman & Lohndal (2008); see Lahne (2009)
- **Check-F, Parse** $\gg$ *Struc** $\gg$ Stay
  analysis of strong and weak pronouns in Cardinaletti & Stark (1999); see Samek-Lodovici (2006)

An example: phonological realization in head chains in Roberts (1997)

(92) **Restructuring and clitic climbing in Italian**:
Gianni li$_1$ vuole tutti leggere t$_1$
Gianni sie will alle lesen

Generalization:
Infinitives are transparent for extraction if the non-finite verb incorporates into the matrix verb (see Baker (1988)).

Assumption (Roberts (1997)):
Incorporation has in fact taken place in (92); however, it cannot easily be detected because a non-highest head chain member is PF-realized.

(93) a. *[X W$_1$ W$_2$ ], where W$_n$ is a morphological word.

b. A head is spelled out in the highest position of its chain, subject to [(93-a)].

The optimality-theoretic reconstruction is straightforward:

(94) a. *ComplexHead:
   *[X W$_1$ W$_2$ ], where W$_n$ is a morphological word.

b. **Head-Real**:
   A phonologically realized head must not be c-commanded by a copy in its chain.

c. **Ranking**: *ComplexHead** $\gg$ **Head-Real**

**T$_{1k}$: Restructuring and clitic climbing in Italian**

<table>
<thead>
<tr>
<th>Candidates</th>
<th>*ComplexHead</th>
<th>Head-Real</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>O$_1$</strong>: Gianni li$_1$ vuole t$<em>2$ tutti [AGR$</em>{R,P}$ leggere$_2$ t$_2$ t$_1$]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
<tr>
<td><strong>O$_2$</strong>: Gianni li$_1$ vuole leggere$<em>2$ tutti [AGR$</em>{R,P}$ t$_2$ t$_1$]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
<tr>
<td><strong>O$_3$</strong>: Gianni li$_1$ vuole t$<em>2$ tutti [AGR$</em>{R,P}$ t$_2$ leggere$_2$ t$_1$]</td>
<td>![ ]</td>
<td>![ ]</td>
</tr>
</tbody>
</table>

6.3 Optimization in the Minimalist Program

Observation:
At the heart of the minimalist program are elementary operations like Agree, Merge, Move, Delete, Transfer, Select. Given that each operation is supposed to apply as soon as its context for application is present (a general Earliness requirement on derivations), it is clear that there will be conflicts. These conflicts have to be resolved by postulating ranking and minimal violability of constraints.

(95) **Merge before Move** (Chomsky (1995; 2000; 2001; 2005); formulation based on Frampton & Gutmann (1999)):
Suppose that the derivation has reached stage $\Sigma_n$, and $\Sigma_{n+1}$ is a legitimate instance of Merge, and $\Sigma_{n+1}$ is a legitimate instance of Move. Then, $\Sigma_{n+1}$ is to be preferred over $\Sigma_n$. 

(96) **The classic effect:**
   a. There$_1$ seems [TP t$_1$ to be [PP someone$_2$ in the room ]]
   b. *There$_1$ seems [TP someone$_2$ to be [PP t$_2$ in the room ]]

(97) **Optimization of derivational steps**: T as input:
   a. [T to be [PP someone$_2$ in the room ]]
   b. [TP there [T to be [PP someone$_2$ in the room ]] ]
   c. *[TP someone$_2$ [T to be [PP t$_2$ in the room ]] ]

(98) **Avoidance of the effect if no expletive is present in the numeration:**
   a. Someone$_2$ seems [TP t$_2$ to be t$_2$ in the room ]
   b. [TP someone$_2$ to be t$_2$ in the room ]

(99) **A potential problem for Merge before Move, part (i):**
   a. *John$_1$ expected [TP t$_1$ to be a proof$_2$ discovered ]
A potential problem for Merge before Move, part (ii):

- a. *C [TP Was a proof discovered] (Merge)
- b. [CP C [TP A proof was t2 discovered] (Move)

Problem:
Why is Move not blocked by Merge (of the external argument or the complementizer) after the generation of T in (99) and (100)?

Solution:
Merge before Move in (95) demands that the preferred option be legitimate. Merge of the external argument John in (99) violates the Theta-Criterion; and Merge of the complementizer in (100) violates subcategorization requirements of T.

Another potential problem for Merge before Move, part (i):

- a. *It1 seems [CP (that) t1 was told John [CP that Bill left]] (Merge)
- b. Itt1 seems [CP (that) John2 was told [CP that Bill left]] (Move)

Another potential problem for Merge before Move, part (ii):

- a. *It1’s fun [TP t1 to [VP PRO2 go to the beach]] (Merge)
- b. It1’s fun [CP PRO2 to [TP t2 go to the beach]] (Move)
- c. *It1 was decided [CP t1 to [VP PRO2 be executed at dawn]] (Merge)
- d. It1 was decided [CP PRO2 to [TP t2 be executed at dawn]] (Move)

Problem:
Why doesn’t the first example in (101) and in (102) block the second one because of Merge before Move? Here, the expletive must be part of the numeration.

Chomsky’s (2000) solution via lexical subarrays:

Suppose ... that at each stage of the derivation a subset LAi is extracted, placed in active memory (the ‘work space’), and submitted to the procedure L. When LAi is exhausted, the computation may proceed if possible. Or it may return to LA and extract LAi, proceeding as before. The process continues until it terminates. Operative complexity in some natural sense is reduced, with each derivation accessing only part of the LA. If the subarray in active memory does not contain EXPL, then Move can take place in the corresponding stage; if it does, Merge of EXPL preempts Move. The next step is to determine the subarrays LAi that can be selected for active memory. LA should determine a natural syntactic object ... the counterpart to a proposition. ... LAi can then be selected straightforwardly: LAi contains an occurrence of C or of v ... — exactly one occurrence if it is restricted as narrowly as possible.

Chomsky (2000)

Thus, each LAi corresponds to a phase.

Consequence:
In there-constructions (where Merge before Move effects obtain), the expletive and the DP always have to be in the same LAi. In it-constructions (where there are no Merge before Move effects), the expletive and the DP do not have to be in the same LAi; they can be in different subarrays.

Optimality-theoretic reconstruction:

Another example (Chomsky (2001)):
The INCLUSIVENESS condition must be minimally violable in favour of the requirement that intermediate steps of successive-cyclic movement proceed via edge feature insertion: Edge features on phase heads are not part of the numeration.

7. Abstractness of Optimality-Theoretic Syntax

Example:
Legendre, Smolensky & Wilson (1998) on accounting for wh-island effects without relying on the concept of intervention (as in Rizzi (1990; 2004)).
ment is that a V that selects an interrogative CP finds a wh-item in the embedded SpecC (or C) position.

(104) Wh-Islands: Optimality Theory:
   a. How do you think [CP Mary fixed the car t₁]?
   b. *How does she know [CP [DP which car] Mary fixed t₂ t₁]?

Consequence:
(i) Extraction of the adjunct violates locality constraints on movement (Loc) in exactly the same way in (104-a) and (104-b): There is no intervention effect induced by which car in (104-b).
(ii) (104-b) is ungrammatical because it blocked by a competing candidate with a better constraint profile: (105) violates the constraint that would normally trigger wh-movement to the matrix clause (Wh-Crit), but since this constraint is ranked lower than the locality constraint violated with extraction from all CPs (Loc), (105) is the optimal candidate.

T₅₁: Wh-islands for long wh-movement of adjuncts

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>SEL</th>
<th>LOC</th>
<th>Wh-Crit</th>
</tr>
</thead>
<tbody>
<tr>
<td>C₁: how₁ C₁ [+wh] ... V [+wh] [CP t₁' ... t₁ ... ]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[+wh] C₂: C₁ [+wh] ... V [+wh] [CP how₁ ... t₁ ... ]</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(105) Optimal candidate blocking wh-movement from wh-island:
She knows [CP how₁ [DP which car₂] Mary fixed t₂ t₁]

Problem:
(105) does not look like a well-formed English sentence.

Solution:
Given the copy theory of movement, wh-in situ may involve multiple wh-movement with selective PF realization of copies, such that one of the traces (rather than its antecedent) is PF-realized in English (Pesetsky (2000), Fanselow & Čavar (2001), Grewendorf (2001)).

Problem:
Why is (104-a) possible after all?

Solution:
The matrix verb in (104-a) selects a declarative CP complement, and not an interrogative CP complement; but if the wh-phrase how stays in the embedded SpecC position, the embedded clause will have to be interpreted as a wh-clause. This would violate highest-ranking SEL.

Note:
A third candidate in which the wh-phrase stays in situ throughout the derivation must also be considered. As a matter of fact, as it stands, this output O₃ would qualify as optimal in

T₅₂: Transparent declarative clauses

<table>
<thead>
<tr>
<th>Kandidaten</th>
<th>SEL</th>
<th>LOC</th>
<th>Wh-Crit</th>
</tr>
</thead>
<tbody>
<tr>
<td>[CP how₁ C₁ [+wh] ... V [+wh] CP t₁' ... t₁ ... ]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[CP how₂ ... t₂ ... ]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>