Week 12: Optimality Theory

Reading: Archangeli chapter

Optimality theory does not necessarily change our representational devices or levels in phonology. Serious reexamination of rules, still within generative phonology framework.

Kisseberth (1970): shows Yawlemani has a conspiracy of rules, resulting in permissible syllables for the language.

Historically in Phonology, we wrote rules of the form:
\[ A \rightarrow B / C D \]

But as a theory of Structural Descriptions (CAD) and operations of Structural Change (A→B), it missed an object of study: the character of the output (CBD)

OT characterizes the output as a result of a struggle between markedness and faithfulness constraints.

**Syllable structure:**
- most unmarked: CV (Hawaiian, also allows slightly marked V)
- more marked: CVC (Yawelmani. also allows unmarked CV)
- much more marked: (C)(C)(C)V(C)(C) (English)

**Yawelmani**
CV(X) syllables explain all the possible sequences found in words, and disallows all the impossibilities (note: words have no appendix).

Rather than rules that accidently conspire to give the right output, use the constraints directly in phonology.

In Yawelmani: ONSET, PEAK, *COMPLEX always true on the surface, but NOCODA sometimes not

**OT Mechanics and Representations**
Grammar defines a relation between input and output, pairs of underlying and surface forms: \( (input_i, output_j) \)

\[ \text{GEN}(\text{Input}_k) \rightarrow \{ \text{Out}_1, \text{Out}_2, \ldots \} \quad \text{(part of UG)} \]

Gen creates infinite possible outputs for a given input form.

\[ \text{H-EVAL}(\text{Out}_j, 1 < j < \infty) \rightarrow \text{Out}_{real} \]

H-eval picks out the best as the real output.

The burden of explanation in phonological theory then lies in H-EVAL, which determines the "best"; the workings of H-eval are shown in a "tableau"
Formalism: The "tableau"

- $C_1 , C_2 , C_3$ are constraints; here $C_1$ outranks $C_2$ outranks $C_3$, and the higher ranked constraint is always written to the left.
- When a possible output violates a constraint, an * appears in the cell below that constraint, in the row of the candidate output.
- The "!" indicates a fatal violation; the candidate output loses because of that violation.
- The $\Rightarrow$ points to the winning output, the one with fewer or lower ranked violations.
- No number of violations of a lower ranked constraint is as bad as a violation of a higher ranked constraint. Think alphabetical order. Azure comes before Babar.
- Shading indicates that a cell no longer contributes to the evaluation; that candidate has already won or lost.

Tableau 1: Optimality Evaluation Schema

<table>
<thead>
<tr>
<th>Candidate Outputs</th>
<th>$C_1$</th>
<th>$C_2$</th>
<th>$C_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>output$_1$</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\Rightarrow$ output$_2$</td>
<td></td>
<td>*</td>
<td>***</td>
</tr>
</tbody>
</table>

Principles of OT  
(McCarthy and Prince 1993a: p. 4, and fn 3)

(1)  

(a) Violability  
Constraints are violable; but violation is minimal.

(b) Ranking  
Constraints are ranked on a language-particular basis; the notion of minimal violation is defined in terms of this ranking (lexico-graphically).

(c) Inclusiveness  
The constraint hierarchy evaluates a set of candidate analyses that are admitted by very general considerations of structural well-formedness.

(d) Parallelism  
Best satisfaction of the constraint hierarchy is computed over the whole hierarchy and the whole candidate set. There is no serial derivation.

References  
(Archive address: http://roa.rutgers.edu/)


