Targeted constraints (Colin Wilson, UCLA)

Orderings
1. An ordering is a relation that is transitive (if \(aRb\) and \(bRc\), then \(aRc\)), and either
   - reflexive (for all \(a, aRa\)) and antisymmetric (if \(aRb\) and \(bRa\) then \(a=b\))
   OR
   - irreflexive (for all \(a, \neg aRa\)) and asymmetric (if \(aRb\) then \(\neg aRa\))

2. OT constraints as traditionally conceived impose an order of the second type, where the relation is “is more harmonic than”:
   Transitive
   - If candidate \(a\) is more harmonic than \(b\) (with respect to some constraint \(C\)), and \(b\) is more harmonic than \(c\), then \(a\) is more harmonic than \(c\).

   Irreflexive
   - A candidate can’t be more harmonic than itself.

   Asymmetric
   - If \(a\) is more harmonic than \(b\), then \(b\) can’t be more harmonic than \(a\).

3. Moreover, the constraint imposes an equivalence relation, “is as harmonic as”, which is transitive, reflexive, and symmetric:
   Transitive
   - If candidate \(a\) is as harmonic as \(b\), and \(b\) is as harmonic as \(c\), then \(a\) is as harmonic as \(c\).

   Reflexive
   - Every candidate is as harmonic as itself.

   Symmetric
   - If \(a\) is as harmonic as \(b\), then \(b\) is as harmonic as \(a\).

4. Finally, for any candidates \(a\) and \(b\), either \(a\) is more harmonic than \(b\), or \(b\) is more harmonic than \(a\), or \(a\) is as harmonic as \(b\).
5. Example of a set of candidates with these two relations imposed (solid arrow = is more harmonic than, dashed arrow = is as harmonic as)

```
  a  c  b
 /    /    /
\   /  /    \
  d  e  f
```

6. A constraint prefers $a$ over $b$ iff $a$ is more harmonic than $b$.

**Targeted constraints**

7. Wilson proposes that all markedness constraints are *targeted*.

8. A targeted constraint prefers $a$ over $b$ iff
   - $b$ violates the constraint
   - $A$ is the set of all candidates that are more harmonic than $a$ in the traditional sense
   - $a$ is a member of $A$ such that no other member of $A$ is more similar to $b$ than $a$ is.

   Where there is some universal map from which similarity can be determined (Steriade’s P-map).

9. Otherwise, the constraint has no preference between $a$ and $b$.

10. A transitive, irreflexive, asymmetric relation “is more optimal than” gets slowly built up as we move from higher-ranked to lower-ranked constraints:
   - If highest-ranked constraint prefers candidate $x$ over $y$, then $x$ is more optimal than $y$.
   - If next-highest constraint prefers $w$ over $z$, then $w$ is more optimal than $z$, unless the previous constraint has already established the opposite.
   - Repeat previous step.

   At the end, there should be one candidate left that is more optimal than every other candidate.

11. Let’s go through an example from Wilson’s handout.
Implications for Samek-Lodovici & Prince’s Optima

12. How do we redefine ‘favor’, since the constraints don’t divide candidates up into neat strata?

13. What implications does this have for the rest of the definitions and for the algorithm?

Implications for Primitive OT

14. Adding constraints and using BestPaths to make ever-more-restrictive FSAs won’t do the job, since we don’t just whittle away at a candidate set.

15. The set does get smaller after certain constraints: if \( x \) is more optimal than \( y \), then \( y \) won’t be the winner.

16. But we can’t just throw away \( y \), because it may be the crucial link to establish by transitivity that \( x \) is more optimal than some other candidate \( w \).

17. So we have to keep track of the “is more optimal than” relation.

18. Can we think of finite-state ways of doing this? What are the complexity implications?

Sympathy

The Proposal

19. Opacity effects can be achieved through constraints that require faithfulness to nonoptimal candidates.

20. For some faithfulness constraint \( F \) (the flower-designating constraint or selector), out of the set of candidates that satisfy \( F \), pick the optimal candidate (flower candidate).

21. Another faithfulness constraint (the flower constraint) requires faithfulness to the flower candidate.

22. Example: /deʃ?/ \( \rightarrow \) [deʃe] (why epenthesis if no surface consonant cluster?)

- The flower-designating constraint is MAX-C.
- The best candidate that satisfies MAX-C is \( \ominus [deʃe?] \)
- The flower constraint is \( \oplus \text{MAX-V} \), requiring the flower candidate’s vowel to be preserved, and it prefers [deʃe] over [deʃ].
Implications

23. Grammar must be run once for each selector—no big problem, though it slows us down if there are a lot of selectors.

24. Then the grammar can be run for real, but the flower candidates need to be somehow stored for evaluation of the flower constraints.

25. Can we “store” the flower candidates by rewriting the flower constraints as FSAs that require faithfulness to that particular representation?

Other topics

Constraint conjunction

26. Does the learner have to consider and rank all conjoined constraints? Are they universal?

Cyclic OT

27. Effect doesn’t seem too big: you just have to run the grammar a few times.

Others you want to talk about?

Next time

• OT and connectionism

For next time

• Read Smolenksy’s Grammar-based Connectionist Approaches to Language (http://128.220.29.11/pdf/grammar-based_connectionist_approaches_language.pdf)