Variations in Optimality Theory: Simulated Annealing and other methods

Tamás Bíró, Dicky Gilbers, Maartje Schreuder

Center for Language and Cognition Groningen
University of Groningen

TABU Day
June 3, 2005
Optimality Theory: a model of language typology

- Candidates = types of language
- Constraint hierarchy
- Thus: one output per hierarchy
- \[ mɛlk \sim [mɛlək] \]: how to account for variations?
Overview


- An alternative proposal: Simulated Annealing for Optimality Theory [Bíró, forthcoming]

- The adequateness of Simulated Annealing for Optimality Theory

- Conclusion
More outputs in Optimality Theory?

For one hierarchy, exactly one violation profile can be optimal.

- One violation profile = more candidates
- One grammar = more hierarchies
- Output = not only the optimal one
**Option 1: one grammar = more hierarchies (1)**

1/1: Ad hoc (?) reranking of the constraints,

E.g. [Schreuder and Gilbers(2004)]:

<table>
<thead>
<tr>
<th></th>
<th>OOC</th>
<th>*ΣΣ</th>
<th>PRS-σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>fototoestel</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Slow (andante) speech:

<table>
<thead>
<tr>
<th></th>
<th>OOC</th>
<th>*ΣΣ</th>
</tr>
</thead>
<tbody>
<tr>
<td>(fóto)(tòestel)</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>(fóto)toe(stèl)</td>
<td>!</td>
<td>*</td>
</tr>
</tbody>
</table>

### Fast (allegro) speech:

<table>
<thead>
<tr>
<th></th>
<th>OOC</th>
<th>*ΣΣ</th>
</tr>
</thead>
<tbody>
<tr>
<td>(fóto)(tòestel)</td>
<td></td>
<td>!</td>
</tr>
<tr>
<td>(fóto)toe(stèl)</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

- Q: Categorical switch to a totally new grammar?
Option 1: one grammar = more hierarchies (2)

1/2: Unordered constraints [Anttila and Cho(1998)]
All permutations yield their output with equal probability

<table>
<thead>
<tr>
<th>fototoestel</th>
<th>OOC</th>
<th>*ΣΣ</th>
<th>PRS-σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>(fóto)(tôestel)</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>(fóto)toe(stèl)</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

- Simple.
- Really free alternations (no control)
- Only certain probability distributions are possible: 50% each.
Option 1: one grammar = more hierarchies (3)

1/3: Stochastic reranking

[Boersma and Hayes(2001)]

Robust learning algorithm (GLA)

- Most probability distributions possible:
- Two constraints: alternative forms < 50%.
- Same probability of reranking, independently of input.
Option 2: Candidates with the same violation marks

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>...</th>
<th>Ci</th>
</tr>
</thead>
<tbody>
<tr>
<td>cand1</td>
<td>*</td>
<td>*</td>
<td>...</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>cand2</td>
<td>*</td>
<td>*</td>
<td>...</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>cand3</td>
<td>**!</td>
<td></td>
<td>...</td>
<td></td>
<td>**</td>
</tr>
<tr>
<td>cand4</td>
<td>*!</td>
<td>*</td>
<td></td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

- There is always a constraint differentiating between them:
- even if low ranked, the analysis does not work.
- Very low ranked constraints are inactive?
Option 3: Non-optimal also as outputs (1)

3/1: critical constraint [Coetzee(2004)]

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>...</th>
<th>C-crit</th>
<th>Ci</th>
</tr>
</thead>
<tbody>
<tr>
<td>cand1</td>
<td>*</td>
<td>*</td>
<td>...</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>cand2</td>
<td>*</td>
<td>*</td>
<td>...</td>
<td></td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>cand3</td>
<td>**!</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td>**</td>
</tr>
<tr>
<td>cand4</td>
<td>*!</td>
<td>*</td>
<td>...</td>
<td></td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

- All candidates surviving the critical constraint appear as alternative forms.

- Non-optimal candidates also ranked: better $\rightarrow$ more frequent.
Option 3: Non-optimal also as outputs (2)

3/2: simulated annealing [Bíró, forthcoming]

- Neighbourhood structure on the candidate set

- Vertical dimension = harmony; random walk in this landscape

- If neighbour more optimal: move.

- If less optimal: move in the beginning, don’t move later
Option 3: Non-optimal also as outputs (3)

3/2: simulated annealing [Bíró, forthcoming]

- Neighbourhood structure $\rightarrow$ local optima
- System can get stuck in local optima: alternation forms
- Precision of the algorithm depends on its speed.
- Alternation forms with $>50\%$ probability can be accounted for.
The Art of Using Simulated Annealing Optimality Theory

• Take a traditional OT model

• Add some convincing neighbourhood structure to the candidate set

• Local (non-global) optima = alternation forms

• Run simulation (some more technical details needed...):
  — Slowly: very likely to be returned only the grammatical form
  — Quickly: quite likely to be returned local (non-global) optima
Is Simulated Annealing Better?

<table>
<thead>
<tr>
<th>fototoestel</th>
<th>OOC</th>
<th>*ΣΣ</th>
<th>PRS-σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>(fóto)(tòestel)</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>(fóto)toe(stèl)</td>
<td>*!</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Problems with different approaches:

- Categorical reranking: both forms present at higher speech rates?
- Unranked constraints (Anttila): speech rate differences?
- Stochastic OT (Boersma): $>50\%$ fast speech forms?
- Coetzee: OOC is the critical constraint, thus all forms are $\sim$!
- All approaches: different input – same probability of reranking
Simulated Annealing Is Better!

\[\text{uit.ge.ve.rij vs. uit.ge.ve.rij}\]

\[\text{stu.die.toe.la.ge vs. stu.die.toe.la.ge}\]

\[\text{per.fec.tio.nist vs. per.fec.tio.nist}\]

[Schreuder (forthcoming) vs. Bíró (forthcoming)]

First two: slow (andante) speech vs. last two: fast (allegro) speech
Simulated Annealing Is Better!

<table>
<thead>
<tr>
<th>fo.to.toe.stel</th>
<th>uit.ge.ve.rij</th>
<th>stu.die.toe.la.ge</th>
<th>per.fec.tio.nist</th>
</tr>
</thead>
<tbody>
<tr>
<td>'camera'</td>
<td>'publisher'</td>
<td>'study grant'</td>
<td>'perfectionist'</td>
</tr>
<tr>
<td>OOC to: susu</td>
<td>ssus</td>
<td>susuu</td>
<td>usus</td>
</tr>
<tr>
<td>fó.to.tè.stel</td>
<td>ùit.ge.ve.ríj</td>
<td>stú.die.tè.lá.ge</td>
<td>per.fèc.tio.níst</td>
</tr>
<tr>
<td>slow: 1.00</td>
<td>slow: 0.97 / 0.96</td>
<td>slow: 0.96 / 0.81</td>
<td>slow: 0.91 / 0.20</td>
</tr>
<tr>
<td>fast: 0.82</td>
<td>fast: 0.65 / 0.67</td>
<td>fast: 0.55 / 0.38</td>
<td>fast: 0.49 / 0.13</td>
</tr>
<tr>
<td>fó.to.toe.stèl</td>
<td>ùit.ge.ve.ríj</td>
<td>stú.die.toe.là.ge</td>
<td>pèr.fèc.tio.níst</td>
</tr>
<tr>
<td>slow: 0.00</td>
<td>slow: 0.03 / 0.04</td>
<td>slow: 0.04 / 0.19</td>
<td>slow: 0.07 / 0.80</td>
</tr>
<tr>
<td>fast: 0.18</td>
<td>fast: 0.35 / 0.33</td>
<td>fast: 0.45 / 0.62</td>
<td>fast: 0.39 / 0.87</td>
</tr>
</tbody>
</table>

Simulated and observed frequencies.

[Bíró (forthcoming), Schreuder (forthcoming)]
Summary

• Different approaches to variation in OT

• Simulated Annealing is a promising alternative

• OT = competence model: what is an optimal candidate

• SA = performance model: finds optimal candidate
  - A near-good solution returned in constant time
  - You can speed up the algorithm by decreasing precision (cf. fast speech phenomena)
Thank you for your attention!
Bibliography

References


[Schreuder and Gilbers(2004)] Maartje Schreuder and Dicky Gilbers. The influence of speech rate on rhythm patterns. In Dicky Gilbers, Maartje