A Computational Account of Tone Sandhi Interaction

Jane Chandlee

Tri-Co Department of Linguistics
Haverford College

AMP 2018 - UCSD
6 October 2018

Complex interaction among three rules that differ in directionality ("confused traffic").

Will present a unified computational account: these rules share the property of being (Right) Output Strictly Local functions.
Rules/processes as maps

- Phonological generalizations (whether stated with rules or ranked constraints) can be viewed as maps (or functions) from an input string to an output string.
- Question of computational interest: how powerful/expressive do these maps have to be?
- Classification of phenomena in terms of *subregular* function classes: ISL and OSL
Input Strict Locality (ISL)

- Output is determined based on a ‘window’ of the most recent input.

\[
\begin{array}{c}
\checkmark \text{ISL} \\
\times \quad \text{Np} \quad a \quad s \quad e \quad b \quad \leftarrow \quad \times \\
\leftarrow \quad \lambda \quad \text{mp} \quad a \quad s \quad e \quad b \\
\times \quad b \quad a: \lambda \quad t \quad \checkmark \\
\end{array}
\quad \quad \quad \quad \quad \quad
\begin{array}{c}
\neg \text{ISL} \\
\times \quad s \quad a \quad a \quad a \quad \leftarrow \quad \times \\
\leftarrow \quad s \quad a \quad a \quad a \quad s \\
\times \quad b \quad a: \lambda \quad t \quad \checkmark \\
\end{array}
\]
Output Strict Locality (OSL)

- Output is determined based on a ‘window’ of the most recent output.

<table>
<thead>
<tr>
<th>Left-OSL</th>
<th>Right-OSL</th>
</tr>
</thead>
<tbody>
<tr>
<td>× N V V V V C ×</td>
<td>× C V V V V N ×</td>
</tr>
<tr>
<td>N ĭ ĭ ĭ ĭ ĭ C</td>
<td>C ĭ ĭ ĭ ĭ ĭ N</td>
</tr>
</tbody>
</table>
Rule application

<table>
<thead>
<tr>
<th>ISL</th>
<th>≈ simultaneous application</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Diagram](ISL simultaneous application)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OSL</th>
<th>≈ iterative application</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Diagram](OSL iterative application)</td>
<td></td>
</tr>
</tbody>
</table>
Tianjin Tone Sandhi

- Four tones: L, H, R=LH, F=HL
- Sequences of 2 identical tones undergo sandhi (except for HH)

(1)  
   a.  FF $\mapsto$ LF  
   b.  LL $\mapsto$ RL  
   c.  RR $\mapsto$ HR
Tritonal strings

\[(1a) \quad FF \leftrightarrow LF \quad (1b) \quad LL \leftrightarrow RL \quad (1c) \quad RR \leftrightarrow HR\]

- Apply right-to-left

\[
\begin{align*}
(1a) & \quad \begin{array}{c}
\text{F} \\
\downarrow \\
\text{FLF}
\end{array} \\
(1b) & \quad \begin{array}{c}
\text{L} \\
\downarrow \\
\text{LRL}
\end{array}
\end{align*}
\]
Tritonal strings

\( (1a) \) FF ↔ LF \hspace{1cm} \( (1b) \) LL ↔ RL \hspace{1cm} \( (1c) \) RR ↔ HR

- Applies left-to-right
Rule-based accounts

Zhang (1987):
- FF rule feeds LL rule
- LL rule feeds RR rule

<table>
<thead>
<tr>
<th>Rule</th>
<th>Direction</th>
<th>RLL</th>
<th>LFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>F→L / F</td>
<td>right-to-left</td>
<td>–</td>
<td>LLF</td>
</tr>
<tr>
<td>L→R / L</td>
<td>right-to-left</td>
<td>RRL</td>
<td>RLF</td>
</tr>
<tr>
<td>R→H / R</td>
<td>left-to-right</td>
<td>HRL</td>
<td>–</td>
</tr>
</tbody>
</table>

See also Tan (1987)
Hung (1987):

- Rules apply freely subject to a phonotactic constraint against adjacent low tones (HH is okay).
- The constraint both ‘positively motivates’ and ‘negatively motivates’ FF $\mapsto$ LF and LL $\mapsto$ RL

\[
\begin{array}{c}
\text{F FF} \\
\downarrow \\
\text{FLF}
\end{array}
\quad
\begin{array}{c}
\text{FF F} \\
\downarrow \\
\text{L FF} \\
\downarrow \\
*\text{LLF}
\end{array}
\]
Constraint-based accounts

- Ad hoc stipulation of directionality seems to be unavoidable in a rule-based approach (Chen 2000, Lin 2008, Wee 2010).
- With violable constraints, sandhi can be assumed to always apply left-to-right unless an OCP violation results.
  - HH is exempt
Constraint-based accounts

Derivational constraints (Chen 2000)
- Temporal Sequence: apply rules left to right
- $\text{OCP}_{\neg HH} \gg \text{Temporal Sequence}$
- $\text{GEN}$ produces candidates of the form:

\[
\begin{align*}
\text{R} & \quad \text{LL} \\
\downarrow \\
\text{RR} & \quad \text{L} \\
\downarrow \\
\text{HRL}
\end{align*}
\]
Constraint-based accounts

- The use of derivational constraints recreates the duplication problem that motivated OT in the first place.
- Standard OT has issues with the opaque nature of left-to-right (Lin 2008, Wee 2010).
  - Overapplication: \( \text{RRR} \mapsto \text{HHR} \)
Constraint-based accounts

Lin (2008):

- **IDENT-BO-T**: Corresponding tones in prosodically related bases and outputs must be identical.

- **OCP(L,F,R) \(\gg\) IDENT-BO-T

\[
\begin{align*}
(RR) & \quad ((RR)R) \\
\downarrow & \quad \downarrow \\
(HR) & \quad ((HH)R) \\
\quad \rightarrow & \quad *(RH)R)
\end{align*}
\]
Computational account: FF and LL rules

- FF and LL rules are 1) regressive and 2) iterative
- Right-OSL

**Diagram:**

- FF and LL rules are represented with 4 states: #, L, R, F.
- Transitions are indicated with arrows from one state to another.
- Input symbols: #, L, R, F.

**UR:** (Up-Right) #, L, L, L, #

**SR:** (Start-Right) L, R, L

**UR:** (Up-Right) #, F, F, F, F, #

**SR:** (Start-Right) F, L, F
Computational account: RR rule

\[ R \rightarrow H / \_ \_ R \]

```
\[ \begin{array}{c}
R & R & R \\
\downarrow & & \\
H & R & R \\
\downarrow & & \\
HHR & & \\
\end{array} \]
```

```
\[ \begin{array}{c}
R & R & R \\
\downarrow & & \\
R & H & R \\
\downarrow & & \\
*RHR & & \\
\end{array} \]
```
Computational account: RR rule

\[ R \rightarrow H / \_ \_ R \]

- Direction actually doesn’t matter.

- This is also simultaneous application: so ISL?
It is ISL, but it’s also Right-OSL.

![Diagram showing the computational account of RR rule]

- UR: # R R R #
- SR: H H R
What’s going on?

- ISL (unlike OSL) can model rules with two-sided contexts.
- When the rule has a one-sided context, the ISL/OSL distinction sometimes collapses.
- Conjecture (for simultaneous application):
  - $A \rightarrow B / \_ \_ D$ (ISL = ROSL)
  - $A \rightarrow B / C \_ \_ \_ \_ (ISL = LOSL)$
- Dissimilation only?
Since they all have the same property (ROSL), the rules can actually be modeled with a single ROSL function.

Advantage: all three can be learned by the OSL learning algorithm (Chandlee, Eyraud, Heinz 2015).
3 rules, single 2-ROSL function
3 rules, single 2-ROSL function
The assumption of 3 rules originally simplified the rule-based analyses.

“There is an implicit assumption in Chen’s [1986] analysis...that there is only one tone sandhi rule in Tianjin, albeit with four subrules, and the four subrules behave identically with respect to the domain and mode of application, with no ordering among them” (Zhang 1987, p. 251).
As originally defined, ISL = simultaneous application and OSL = iterative application, but it’s more nuanced than that.

- Under certain conditions, OSL can model either mode of application.

A unified analysis of Tianjin Tone Sandhi is possible: all three rules are ROSL.

- Exceptionality of the RR rule is only apparent.
- Though it is still distinct in also being ISL.
Tone sandhi provides useful case studies for the computational nature of different modes of application:

- Strings of identical targets (RRR, FFF, LLL)
- Target/trigger overlap among rules.

Are there other cases of sandhi interaction in which the rules are (necessarily) computationally distinct?


