1. Introduction

Parallelism as central tenet of Optimality Theory (Parallel OT; Prince & Smolensky 1993/2004):
- Evaluation is over all candidates, including those displaying multiple changes from input.
- Phenomena argued to support parallelism:
  - Top-down & bottom-up interactions (Prince & Smolensky 1993/2004)
  - Copying and markedness repair (McCarthy & Prince 1995)

Harmonic Serialism (HS; McCarthy & Pater 2016) adopts constraints, abandons parallelism:
- Evaluation only over candidates whose output differs from input by the application of at most one change (“operation”; McCarthy 2016).
- Overcomes certain theoretical consequences of Parallel OT:
  - Co-optima problems (McCarthy 2008)
- Can derive phenomena previously argued to vindicate parallelism:
  - Top-down & bottom-up interactions (McCarthy, Pater, & Pruitt 2016)
  - Copying and markedness repair (McCarthy, Kimper, & Mullin 2012)

Can we find better support for parallelism?

McCarthy (2013) sets a clear criterion for definitive support for parallelism (1).

(1) Def’n: Let \( x \) be an input and \( op_1 \) and \( op_2 \) be distinct operations. \( op_1 \) and \( op_2 \) are **irreducibly parallel** if \( \text{GEN} \) must generate a candidate \( op_2(op_1(x)) \) in a single step.

Given the definition of (1), this paper has two goals:

(2) Elucidate the ranking conditions under which generation of \( op_2(op_1(x)) \) is necessary.

(3) Demonstrate that a diverse set of attested phenomena meet these conditions.
2. Foot binarity in Mohawk

In Mohawk, different sets of processes apply to ensure that feet are bimoraic.
- These interactions are successfully derived in parallel OT, but not HS (Adler 2016).
- To express these interactions, footing and lengthening must apply in parallel.

2.1 Data (Michelson 1988)


(4) In closed syllables, foot is always monosyllabic: (‘CVC), or (‘CeC)


b. For epenthetic e: (‘CeC) /wak-nyak-s/ [.wa(‘.ken.)yaks.] 1P-marry-HAB I marry

(5) In open syllables however, foot form varies: (‘CV:) or (‘CV.Ce)

a. For underlying V: (‘CV:) , *(‘CV.CV) /k-haratat-s/ [.kha(‘.ra:.tats.] 1A-lift-HAB I am lifting it up a little

b. For epenthetic e: (‘CV.Ce), *(‘Ce:) /w-akra-s/ [(.wa.ke:)ras.] NA-smell-HAB It smells

In sum: In open syllables, (‘CV:) or (‘CV.CV) are possible feet.
➢ (‘CV:) generally preferred to (‘CV.CV). (6a)
➢ (‘CV.Ce) preferred to (‘Ce:) to avoid long e. (6b)

2.2 Constraints:

(6) a. FtbIN Assign a violation for a foot containing more or less than two morae.

b. DEpµ Assign a violation for vowel lengthening.

c. IAMb Assign a violation for each foot of the form (‘σ σ).

d. DEpV: Assign a violation for each long epenthetic vowel.

2.3 Mohawk in Parallel OT: the successful derivation

In non-epenthetic environments, two rankings account for lengthening:
- FtbIN >> DEpµ: Vowel lengthening occurs to ensure foot binarity. (7a, b)
- IAMb >> DEpµ: Vowel lengthening is preferred over building a trochaic foot. (7a, c)
In epenthetic environments, one ranking accounts for emergence of disyllabic foot:

- \( \text{DepV} : \gg \text{IAMB} \): A disyllabic foot is better than a long epenthetic vowel. (8)

2.4 Mohawk in HS: the failed derivation

In addition to the constraints in 2.2, we define the set of operations as in (10):

(10) a. \( \text{op}_1 : \sigma \rightarrow (\sigma) \) \textbf{Build monosyllabic foot}
b. \( \text{op}_2 : \text{V} \rightarrow \text{V} : \) \textbf{Lengthen vowel}
c. \( \text{op}_3 : \sigma \sigma \rightarrow (\sigma \sigma) \) \textbf{Build disyllabic foot}

2.4.1 The derivation of non-epenthetic forms

For input /CVCVCV/, \( G_{\text{EN}} \) cannot immediately generate \([\text{CV}('\text{CV}:)\text{CV}]\).

- Step 1: /CVCVCV/ \( \rightarrow [\text{CV}('\text{CV}:)\text{CV}] \) \textit{Build Monosyllabic Foot} 
- Step 2: /CV('CV):CV/ \( \rightarrow [\text{CV}('\text{CV}:)\text{CV}] \) \textit{Lengthen vowel}

Step 1

\([\text{CV}('\text{CV})\text{CV}] \sim *[('\text{CV.CV})\text{CV}] \) entails: \( \text{IAMB} \gg \text{FtBin} \). (11)

- \([\text{CV}('\text{CV})\text{CV}] \) must beat *[('\text{CV.CV})\text{CV}], or we fail to derive [CV('CV:)CV].
- But, [CV('CV)CV] doesn’t satisfy FtBin, while *[('CV.CV)CV] does!

<table>
<thead>
<tr>
<th>/CVCVCV/</th>
<th>DepV:</th>
<th>FtBin</th>
<th>IAMB</th>
<th>Depµ</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. ( \infty ) CV('CV):CV</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. ('CV.CV)CV</td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Elfner (2016) offers alternative analysis of Mohawk in HS. See appendix 2 for arguments against her analysis.
Step 2

\text{FtBIN} >> \text{DEPM} \text{uje} \text{s}e \text{h}es \text{e} \text{d} \text{e} \text{r} \text{e} \text{s} \text{d} \text{e} \text{r} \text{a} \text{n} \text{t} \text{c} \text{a} \text{n} \text{d} \text{i} \text{d} \text{e} \text{s}, \text{[CV(‘CV):CV]}]. (12)

- The derivation converges in the following step (not shown), on the attested form.

\[
\begin{array}{|c|c|c|c|}
\hline
/\text{CV(‘CV):CV}/ & \text{DEPV} : & \text{IAMB} & \text{FtBIN} & \text{DEPM} \\
\hline
\text{CV(‘CV):CV} & * & & & \\
\text{CV(‘CV):CV} & * & & & \\
\hline
\end{array}
\]

(12) a. CV(‘CV):CV *!

The derivation of non-epenthetic forms converges on attested candidate, but…

★ What are the consequences of IAMB >> FTBIN \text{for the entire Mohawk conspiracy?}

2.4.2 The derivation of epenthetic forms

Step 1: Build trochee

IAMB >> FTBIN wrongly chooses *[CV(‘Ce)rV]. (13)

- Because FTBIN was demoted, a monosyllabic foot beats a bimoraic foot.

\[
\begin{array}{|c|c|c|c|c|}
\hline
/\text{CVCe}rV/ & \text{DEPV} : & \text{IAMB} & \text{FtBIN} & \text{DEPM} \\
\hline
\text{CVCe}rV & * & & & \\
\text{CVCe}rV & * & & & \\
\hline
\end{array}
\]

(13) a. CVCe rV *

In following steps, derivation will converge on *[CV(‘Ce)rV] or *[CV(‘Ce):rV] (not shown).

2.5 Discussion

(14) Non-epenthesis environments

<table>
<thead>
<tr>
<th>Procedure A</th>
<th>Procedure B</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{op}_1: \text{build monosyllabic foot}</td>
<td>\text{op}_3: \text{build disyllabic foot}</td>
</tr>
<tr>
<td>\text{op}_2: \text{lengthen vowel}</td>
<td>bimoraic foot</td>
</tr>
</tbody>
</table>

★ In Mohawk conspiracy on bimoraic feet, footing and lengthening are irreducibly parallel.

- Where \text{op}_1 = \text{monosyllabic foot}, \text{op}_2 = \text{lengthening}, \text{op}_3 = \text{disyllabic foot}, for an input, \text{x}, \text{GEN} must be able to generate \text{op}_2(\text{op}_1(\text{x})), so it can be compared against \text{op}_3(\text{x}).
3. Irreducible parallelism in conspiracies of procedures

We define *conspiracy of procedures* as follows.

To best satisfy the same set of driving constraints:

- Apply Procedure A, consisting of two single changes in succession:

  \[(15a) \quad /x/ \rightarrow A(x) \rightarrow A(x)\]

  - … unless the result is a marked structure.

- in which case apply Procedure B, which does not share the same first change with Procedure A.

  \[(15b) \quad /x/ \rightarrow B(x)\]

Conspiracy arises if one input undergoes Procedure A, and a different input undergoes B.

⭐ In sum, two distinct *sequences of changes* are compared to best satisfy the set of driving constraints.

The formal expression of a conspiracy of procedures in OT involves the constraints in (16).

(16) a. **DRIVERS**: Assign a violation for some marked structure Satisfied by the application of Procedures A or B

b. ***PROC B**: Violated in the application of some Procedure B repair Accounts for default preference for Procedure A

c. **BLOCKER**: Assigns a violation for a marked result of Procedure A Prefers Procedure B output over Procedure A output

- Mohawk has a single DRIVER, but see Sec. 4 below for case with multiple DRIVERS.

3.2 Conspiracy of procedures in Parallel OT: successful derivation

Suppose inputs \(x\) and \(y\) violate the same DRIVERS and that \(x \rightarrow A(x)\) and \(y \rightarrow B(y)\).

- ***PROC B** prefers Procedure A over Procedure B in satisfying DRIVER. (17b, c)

<table>
<thead>
<tr>
<th>(/x/)</th>
<th>DRIVER</th>
<th>BLOCKER</th>
<th>*PROC B</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td></td>
<td>*!</td>
<td></td>
</tr>
<tr>
<td>A((x))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B((x))</td>
<td></td>
<td></td>
<td>*!</td>
</tr>
</tbody>
</table>
• But higher-ranked BLOCKER prefers B where full result of A yields marked structure. (18b, c)

<table>
<thead>
<tr>
<th></th>
<th>/y/</th>
<th>DRIVER</th>
<th>BLOCKER</th>
<th>*PROC B</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>y</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>A(y)</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>B(y)</td>
<td></td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

**3.2 Conspiracy of procedures in Harmonic Serialism: failed derivation**

**3.2.1 The derivation of Procedure A forms**

- Procedure A: /input/ → A(input) → A(input)
- Procedure B: /input/ → B(input)
- In Step 1, only A and B can be compared.

**Step 1**

Suppose inputs x and y violate the same DRIVERS and we want x → A(x) and y → B(y).

We focus on the driving constraint satisfied by A(x) and B(x) but not by A(x) — in HS, this constraint would drive Procedure B to apply in Step 1, so we call it DRIVERB below.

- A(x) ~ B(x) entails ranking DRIVERB below *PROC B.
- That is, for Procedure A to ever apply, *PROC B >> DRIVERB (19).

<table>
<thead>
<tr>
<th></th>
<th>/x/</th>
<th>BLOCKER</th>
<th>*PROC B</th>
<th>DRIVER</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>A(x)</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>B(x)</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Step 2**

Now A(x) can map to A(x), so that Procedure A forms are successfully derived (20a):

<table>
<thead>
<tr>
<th></th>
<th>/A(x)/</th>
<th>BLOCKER</th>
<th>*PROC B</th>
<th>DRIVER</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>A(x)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>A(x)</td>
<td></td>
<td>*!</td>
<td></td>
</tr>
</tbody>
</table>

**3.2.2 The derivation of Procedure B forms**

**Step 1**

*PROC B >> DRIVERB predicts y maps to *A(y) rather than the desired winner, B(y) (21a).
(21) a.  

<table>
<thead>
<tr>
<th>/y/</th>
<th>BLOCKER: *PROC\textsc{B}</th>
<th>DRIVER\textsc{B}</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>(\text{A}(y))</td>
<td>*</td>
</tr>
<tr>
<td>*!</td>
<td>(\text{B}(y))</td>
<td></td>
</tr>
</tbody>
</table>

In the following step(s), the derivation will land on \(*\text{A}(y)\) or \(*\text{A}(y)\) (not shown).

\textsc{Blocker} only sensitive to full \(\text{A}(y)\), not \(\text{A}(y)\), and so it plays no role in determining outcome.

- HS thus misses generalization that Procedure \textsc{B} applies only when Procedure \textsc{A} results in a marked structure.

### 3.3 Discussion

**In Parallel OT:** constraints can evaluate the results of applying entire procedures at once.

**In HS:** results of applying a given procedure cannot be seen at first step.

- If changes apply one at a time, well-formedness of Procedure \textsc{A} cannot be assessed until after its path is chosen.
- Procedure \textsc{A} operations therefore must apply in parallel in order for blocking to be possible.

General issue does not seem to be addressed by switching from constraint ranking to constraint weighting (Serial Harmonic Grammar; Pater 2012).

- Seemingly only aids when application of Procedure \textsc{B} is successfully reinterpreted as a Step 1 cumulative effect rather than as a Step 2 violation of \textsc{Blocker}.

### 4. Maragoli reduplication-gliding paradox

Conspiracy arises in a reduplication-repair interaction observed in the human possessive paradigm in Maragoli:

- Resolve stem hiatus via glide formation, then copy… \(\text{(23a, 24a)}\)
- unless the result is an additional complex onset… \(\text{(23b, 24b)}\)
- in which case you copy first, then resolve hiatus. \(\text{(23b, 24b)}\)

\begin{align*}
\text{(23a)} & /\text{RED+e+o}/ \rightarrow [\text{j}ɔː-\text{j}-\text{ɔ}] & \text{(23b)} & /\text{RED+vi+o}/ \rightarrow *[\text{vj}ɔː-\text{vj}-\text{ɔ}] \\
& \text{AGR9-\text{your}} & & \rightarrow [\text{vi}-\text{vj}-\text{ɔ}] \\
\text{AGR8-\text{your}} \\
\text{(24a)} & /\text{RED+o+o}/ \rightarrow [\text{wɔː-v}-\text{ɔ}] & \text{(24b)} & /\text{RED+go+o}/ \rightarrow *[\text{gwɔː-gw}-\text{ɔ}] \\
& \text{AGR1-\text{your}} & & \rightarrow [\text{gu}-\text{gw}-\text{ɔ}] \\
\text{AGR3-\text{your}}
\end{align*}
4.1 Maragoli in Parallel OT

(25) **Drivers**: Repair hiatus, and realize reduplicant

- Procedure A: Resolve hiatus and copy result  
  - Operation 1: Glide formation
  - Operation 2: Copy
  results in **full copying**

- Procedure B: Copy first, then resolve hiatus  
  - Operation 2: Copy
  - Operation 1: Glide formation
  results in **partial copying**

Parallel OT treats these data straightforwardly through BR-correspondence (McCarthy & Prince 1995):

(26) **Drivers**: **NoHiatus**: Assign a violation for each hiatus.  
**MAX-BR**: Assign a violation for every base segment that lacks a reduplicant correspondent.

*ProcB*: **MAX-BR**: Disfavors **partial copying**.

**Blocker**: **Complex**: Assign a violation for each complex margin.

Isn’t MAX-BR a driver, not a *ProcB* constraint?
- Indeed, MAX-BR is a driver: it drives copying here.
- But, importantly, it is also a *ProcB* constraint, violated by Procedure B, which displays partial copying.

As illustrated in the tableau below:
- MAX-BR favors full copying… (27b, c)

<table>
<thead>
<tr>
<th>(27)</th>
<th><strong>Driver</strong></th>
<th><strong>Blocker</strong></th>
<th><em>ProcB</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td>RED+e+ɔ</td>
<td>NoHiatus</td>
<td>*!</td>
</tr>
<tr>
<td>b.</td>
<td>eːj-ɔ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.</td>
<td>RED+jːj-ɔ</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Isn’t MAX-BR a driver, not a *ProcB* constraint?
- Indeed, MAX-BR is a driver: it drives copying here.
- But, importantly, it is also a *ProcB* constraint, violated by Procedure B, which displays partial copying.

As illustrated in the tableau below:
- MAX-BR favors full copying… (27b, c)
• but *COMPLEX $\gg$ MAX-BR favors partial copying where full copying would result in an extra complex onset. (28b, c)

\[
\begin{array}{|c|c|c|c|}
\hline
\text{Driver} & \text{Blocker} & *\text{PROC}_B \\
\hline
/\text{RED}+\text{vi}+\text{ɔ}/ & \text{NOHIATUS} & \text{COMPLEX} & \text{MAX-BR} \\
\hline
\text{a.} & \text{RED}+\text{vi}+\text{ɔ} & *! & *** \\
\text{b.} & \text{vi}:\text{vj}:-\text{ɔ} & * & * \\
\text{c.} & \text{vj}:\text{vj}:-\text{ɔ} & **! & \\
\hline
\end{array}
\]

(28)

4.2 Maragoli in HS

Maragoli is challenging for HS: we get copying or glide formation first, but no varying order for purposes of best satisfying prosodic constraints.

Different theory of reduplication in HS: Serial Template Satisfaction (McCarthy, Kimper & McMullin 2012).

(29) \textit{Drivers:} HEADEDNESS: Assign a violation for every headless syllable.
\textit{(drives copying in HS; Selkirk 1995)}

\textit{NoHiatus}

Reminder: Procedure A: Resolve hiatus, then copy
Procedure B: Copy, then resolve hiatus

• \textit{DriverB} is HEADEDNESS, since Procedure B satisfies it in Step 1, but not A.
• \textit{*PROC}_B is NOHIATUS, since Procedure B violates it in Step 1, but not A.
• Once again, \textit{Blocker} is *COMPLEX.

We get ranking paradox between NOHIATUS and HEADEDNESS (HD below):
• (/\text{RED}+\text{e}+\text{ɔ}/ → [jɔː:j-ɔ]) requires NOHIATUS $\gg$ HD so gliding applies first… (30a)

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Step} & /\text{RED}+\text{e}+\text{ɔ}/ & \text{NOHIATUS} & \text{HD} & *\text{COMPLEX} \\
\hline
\text{a.} & 1 & \text{RED}+\text{e}+\text{ɔ} & \text{**!} & \\
\text{b.} & 1 & \text{e}':\text{e}:-\text{ɔ} & *! & \text{*} \\
\text{c.} & 2 & \text{jɔː:j-ɔ} & & \\
\hline
\end{array}
\]

(30)
• but (/RED+vi+ɔ/ → [vi:-vj-ɔ]) requires HD ≫ NOHIATUS, so copying applies first. (31b)

<table>
<thead>
<tr>
<th>Step</th>
<th>/RED+vi+ɔ/</th>
<th>NOHIATUS</th>
<th>HD</th>
<th>*COMPLEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td>1</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>b.</td>
<td>1</td>
<td>vi:-vi-ɔ</td>
<td><em>!</em></td>
<td></td>
</tr>
</tbody>
</table>

• *COMPLEX can play no role in the above,
• and so HS misses generalization that copy and repair apply in onset-optimizing order.

❖ Copying and hiatus repair are thus irreducibly parallel: GEN must be able to generate and compare candidates in which copying and repair apply in the same step.

See Zymet (2015) for more comprehensive discussion.

5. Additional conspiracies of procedures

In our work, we gather the following cases (alphabetically listed by language) and show that they fit into the conspiracies of procedures schema:

<table>
<thead>
<tr>
<th>Language(s)</th>
<th>Driver constraint(s):</th>
<th>do Procedure A…</th>
<th>unless result is…</th>
<th>else do Procedure B:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gurindji</td>
<td>Pre-nasal segments are nasal</td>
<td>Iterative [nasal] spreading</td>
<td>NC₀V sequence</td>
<td>[nasal] deletion</td>
</tr>
<tr>
<td>(Stanton 2016)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lithuanian</td>
<td>Adjacent obstruents agree on [pal] and [voi]</td>
<td>Palatal assim. &amp; voicing assim.</td>
<td>Geminate</td>
<td>[i] epenthesis</td>
</tr>
<tr>
<td>(Bakovic 2005)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maragoli</td>
<td>Reduplicants are realized; no hiatuses</td>
<td>Gliding → reduplication</td>
<td>Complex reduplicant onset</td>
<td>Reduplication → gliding</td>
</tr>
<tr>
<td>(Zymet 2015)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mohawk</td>
<td>Feet are bimoraic</td>
<td>Monosyl. footing → V-lengthening</td>
<td>Long epenthetic vowel</td>
<td>Disyllabic footing</td>
</tr>
<tr>
<td>(Adler 2016)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sino-Jap.</td>
<td>Words are disyllabic, adjacent obstruents agree in place</td>
<td>V-deletion → C-assimilation</td>
<td>Voiced geminate</td>
<td>Nothing</td>
</tr>
<tr>
<td>(Ito &amp; Mester 1996)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

… and the list could expand pending further investigation (Bickmore & Kula 2013, Jardine to appear; Smith 2016; Steriade in prep).

Diversity of phenomena suggests that parallel evaluation is a general property of grammar, rather than being limited to a single exceptional interaction.

See Appendix 1 for more discussion.
6. Outlook and conclusion

Parallel models and serial models of grammar have their own advantages:
- Serial models: avoid too-many-solutions and co-optima problems.
- Parallel models: can derive conspiracies of procedures.

We may need to combine serial & parallel derivation — in OT-CC, for example:
- Operations apply serially, avoiding too-many-solutions effects (McCarthy 2007).
- But entire derivations are compared in parallel, leading to successful treatment of conspiracies of procedures.

Nonetheless, the serial model alone cannot capture conspiracies of procedures.
- Top-down/bottom-up & reduplication-repair interactions gave early support for parallelism
- The necessity of parallelism to derive these phenomena was then cast into doubt
- But, when we look at top-down & reduplication-repair interactions as part of entire phonological conspiracies, they do indeed vindicate parallelism.

Implications of conspiracies of procedures for larger serialism/parallelism debate:
- They require irreducible parallelism in the abstract, free of representational assumptions.
- The diversity of attested cases show that it is not just two specific phonological processes that must run in parallel.
- Rather, these cases suggest speakers are sensitive to the results of whole derivations.

Appendix 1: Additional cases

Assimilation-epenthesis interactions in Lithuanian

Proc. A: Assimilate adjacent stops for voicing and palatality…

\[
/p+d^j/ \rightarrow b+d^j \rightarrow b^j+d^j
\]

unless the result is a geminate…

\[
/p+b^j/ \rightarrow b+b^j \rightarrow \ast b^j+b^j
\]

Proc. B: in which case, epenthesize between the two stops.

\[
/p+b^j/ \rightarrow pi+b^j
\]

Data from Lithuanian on alternations of verbal prefix ap- (Baković 2005, also see Pająk & Baković (2010) for a similar case in Polish):

<table>
<thead>
<tr>
<th>Faithful forms (30a)</th>
<th>Assimilated forms (30b)</th>
<th>Epenthetic forms (30c)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ap-rafi:tii,’describe’</td>
<td>ab-gauti,’deceive’</td>
<td>ap’i-bar’iti (*ab-bar’iti),’spill on’</td>
</tr>
<tr>
<td>ap-tar’iti,’describe’</td>
<td>ap’i-t’em’idi:tii,’obscure’</td>
<td>ap’i-b’er’iti (*ab’-b’er’iti),’strew’</td>
</tr>
<tr>
<td>ab’d’eg’iti,’get’</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

★ Derivations must look ahead to a later stage to determine if NOGEM is violated (Albright & Flemming 2013).
• Couldn’t epenthesis be triggered by a ban on sufficiently similar sequences of obstruents (e.g., /p+b/)?
  o Based on typological evidence Baković (2005) and Pajak & Baković (2010) argue against bans such as these — i.e., for sensitivity only to geminates.

**Deletion-gemination interaction in Sino-Japanese root fusion**

When two CVCV roots are compounded together: /CV_C1VCVC2V/

**Proc. A:** Delete the last V of first root, and assimilate backwards… /tu+k/ → t+k → k+k
  unless the result is a voiced geminate… /ti+b/ → t+b → *b+b
   /tu+n/ → t+n → *n+n

**Proc. B:** in which case, do nothing. /ti+b/ → ti+b
   /tu+n/ → tu+n

The data (Ito & Mester 1996, 2015):

(31a) betu ‘different’
  betu+kaku → bek-kaku ‘different style’
  betu+soo → bes-soo ‘separate mail’
  betu+bin → betu-bin (*beb-bin) ‘separate carrier’
  betu+goo → betu-goo (*beg-goo) ‘separate issue’

(31b) niti ‘sun’
  niti+pi → nip-pi ‘Japan and the Philippines’
  niti+kaN → nik-kaN ‘Japan and Korea’
  niti+bei → niti-bei (*nib-bei) ‘Japan and America’
  niti+maN → niti-maN (*nim-maN) ‘Japan and Manchuria’

★ Derivations must **look ahead to a later stage** to determine if NOVOIGEM is violated.

• Debate over whether the last vowel in Root1 is actually underlying or epenthetic (Ito & Mester 1996, 2015; Kurisu 2000; Kawahara et al. 2003):
  o Quality is mostly predictable, but not in [Cit{i u}] forms.
  o Speakers may not internalize /CVC/ (Kiparsky 1968, Bowers 2012)
  o Impertinent to the question of parallelism/serialism in constraint-based theories that adopt Richness of the Base (Prince & Smolensky 1993/2004).
  o /CVC1VCVC2VCV/ needs an analysis.

**Non-myopic harmony in Gurindji and elsewhere**

**Proc. A:** Spread [nasal] backwards from an NC… /V V +NC/ → V ũ +NC → ũ ũ +NC
  unless result is NCũ… /NC V V +NC/ → NC V ũ +NC → *NC ũ ũ +NC

**Proc. B:** in which case, delete [nasal]. /NC V V +NC/ → NC V V +C
The data (McConvell 1988):

(32) \(/kajira-mpal/ \rightarrow \text{kājirā-mpal} \quad \text{‘across the North’}\\
/kankula-mpa/ \rightarrow \text{kānkula-pa}, (*\text{kānkūlāmpa}) \quad \text{‘on the high ground’}

★ Derivations must look ahead to a later stage to determine if *NCV is violated — see Stanton (2016) for more detailed justification of evidence for parallelism here.

Additional examples of non-myopic harmonies (see Walker 2014 for more discussion):

Appendix 2: Mohawk as a conspiracy on foot binarity

The analysis of Mohawk in §2 rested on two crucial assumptions:

- The Mohawk foot is strictly bimoraic, and is the nexus of the Mohawk stress patterns.
- Vowel epenthesis must precede stress assignment in the derivation of Mohawk within HS.

Elfner (2016) provides an alternative analysis of Mohawk within HS:

- Ignores foot structure for the sake of stress placement
- Emphasizes: 2 processes of [g]-insertion; correlate with different stress patterns.
  - CCC-insertion: Penult stress.
    \(/wak-nyak-s/ \quad [.\text{wa(ˈ.ken.)yaks.}] \quad 1P-\text{marry-HAB} \quad I \text{ marry}\)
  - Cr-insertion: Antepenult stress.
    \(/w-akra-s/ \quad [ˈ.\text{wa.ke}\.ras.] \quad \text{NA-smell-HAB} \quad \text{It smells}\)

- In HS, the constraint ranking determines the order at which operations apply.
  - CCC-insertion >> \text{PenultStress: epenthesis before stress assignment}
  - \text{PenultStress} >> Cr-insertion: stress assignment before epenthesis
    - Gives opaque appearance of antepenult stress.
[e]-forms make it impossible to adjudicate between these two accounts. If we look at other epenthetic processes in the language, however, we see that only the predictions of the output-based account are borne out:

[a]-epenthesis:
- Mohawk insets [a] between the noun and verb in noun incorporation.
  - Single process of epenthesis, driven by single structure in the input.
  - Nonetheless, location of stress varies, depending on syllable structure

(5) In closed syllables, stress is penultimate: CV(‘CaC)CV
/te-wak-iʔtsyuk-nyu-s/ [.te.wa.keʔt.syu(‘kan.)yus.] DU-1P-sneeze-DIST-HAB ‘I’m sneezing’

(6) In open syllables however, stress is antepenultimate: (‘CVC)Ca.CV
/wak-nuhs-yA-Ø/ [.wa.ke(‘nuh.)sA.yA.] 1P-house-put, own-STAT ‘my house’

In sum, [a]-forms display the same alternation as [e]-forms:
- Stress falls on epenthetic vowel [e] or [a] when they occupy a closed penult.
- Stress moves to the antepenult, when [e] or [a] occupies an open penult.
- Thus, the behavior of stress must be a result of syllable structure, not differential ordering between epenthetic processes and a penult stress rule.
- The derivational analysis would be forced to posit two arbitrarily different constraints driving [a]-epenthesis:
  - One constraint ranked above PENULTSTRESS for all cases where [a] receives stress.
  - One constraint ranked below PENULTSTRESS for all cases where [a] does not.
- Hence we need to allow FTBIN to drive the behavior of stress.
  - However for the derivation to ‘see’ all the syllables necessary to make the right computation for the location of stress, the syllable with the epenthetic vowel must be present in the derivation at the point where stress is applied.
- Therefore, epenthesis must always precede foot building.

Key points of this analysis:
- Highlights structures in the input: CCC vs. Cr clusters, to predict stress
- Interprets Mohawk as a case of opacity, an intrinsically serial interaction!
- Differs wholly our analysis:
  - We highlight structures in the output, Syllable Open/Closedness, to predict stress
  - We interpret Mohawk as a conspiracy of procedures, an intrinsically parallel interaction!
References


Albright, Adam & Edward Flemming. 2013. A note on parallel and serial derivations. Class notes, MIT.


Smith, Brian. 2016. Phonologically-conditioned variation in English function words. Talk given at the Southern California Meeting on Phonology.

Stanton, Juliet. 2016. Trigger Deletion in Gurindji. Poster presented at the 24th Manchester Phonology Meeting, Manchester, UK.


Zymet, Jesse. 2015. A case for parallelism: reduplication-repair interaction in Maragoli. Talk presented at the 23rd Manchester Phonology Meeting, Manchester, UK.