Ivy Hauser and Coral Hughto
blogs.umass.edu/ihauser|coralwilliam

Faithfulness-based opacity in Harmonic Serialism

1 Introduction

- Harmonic Serialism (HS), a serial derivative of Optimality Theory, captures some aspects of rule ordering. Processes can apply before others through constraint ranking.
- Some previous work has analyzed opacity in HS (McCarthy 2000, Elfner 2009, Jarosz 2014), but many aspects of opacity have continued to prove problematic.
- **Proposal:** New classes of faithfulness constraints within HS which reference the underlying representation (UR) of forms and/or add a specific context of application to account for opacity.
  - Counterbleeding: Contextual Faithfulness
  - Counterfeeding: $\text{F}\textsc{aith}_{UO}$
- In our paper, we argue that these constraints are induced on a language-specific basis and provide an induction algorithm.
- When a specific inconsistency is detected, the learner is triggered to induce the appropriate constraint. With this system in place, these constraints will only be induced in cases of opacity.

2 Harmonic Serialism

- In HS, GEN is limited to candidates that differ from the input by at most one change (McCarthy, 2010).
- The output of EVAL at one step is the input to the following step.
- The derivation converges when the fully faithful candidate is optimal (no further change is more harmonic).

3 Counterbleeding Opacity

- Counterbleeding opacity results in surface overapplication (Kiparsky, 2000).
- A rule has applied on the surface, but the context for its application is not present.

(1) **Counterbleeding in Arabic:** Palatalization before high vowels, high vowels delete in open syllables; deleted high vowels remove context for palatalization (Al-Mozainy, 2007).

	/ʃaribat/	/ħaːkim/	/ħaːkim-in/
Palatalization		ħaːkʲim	ħaːkʲimin
Deletion	ʃarbat		ħaːkʲmin
	[ʃarbat]	[ħaːkʲim]	[ħaːkʲmin]

3.1 Analyzing Counterbleeding with Contextual Faithfulness

- **Contextual Faithfulness:** Like positional faithfulness, but define an input context, and not limited to prosodically prominent positions

(2) **IDENT(F)/*Context***
If an input segment is [αF] and in context C, then its corresponding output segment must be [αF].

(3) **MAX(*A*)/*Context***
An input segment A in context C must have an output correspondent.

- **EXAMPLE:** For Arabic,

(4) **MAX(i)/k_:** Assign one * if [i] is deleted when preceded by a non-palatalized voiceless consonant in the input.

- The constraint serves to protect the feature or segment until some other process has applied to remove the specified context

3.2 Analysis: Arabic

- **Arabic: Deletion counterbleeds Palatalization**
/haːkim-in/ → haːk^{j}imin → [haːk^{j}min]

(5) **Step 1:** Palatalization occurs

	/haːkim-in/	MAX(i)/k_	*iCV	*ki	IDENT[back]	MAX
→ 1.	haːk^{j}imin		*		*	
2.	haːkmin	*W	L		L	*W
3.	haːkimin		*	*W	L	

(6) **Step 2:** Deletion occurs

	haːk^{j}imin	MAX(i)/k_	*iCV	*ki	IDENT[back]	MAX
→ 1.	haːk^{j}min					*
2.	haːk^{j}imin		*W			L

- The contextual faithfulness constraint prevents the [i] from deleting until its context is no longer met; i.e. until after palatalization has applied.

3.3 Analysis: Double counterbleeding in Yawelmani

- Contextual faithfulness constraints can also analyzed layered overapplication opacity.
- Multiple contextual faithfulness constraints are needed for each individual opaque interaction.
- When all processes can apply to an input, both constraints are active.

(7) Yawelmani ordered processes (Newman, 1944; McCarthy, 2007)

1. *Rounding harmony* (RH): suffix vowel matches rounding to the preceding vowel if they match in height
2. *Lowering* (L): high long vowels become non-high
3. *Closed syllable shortening* (CSS): long vowels shorten in closed syllables

- These three processes combine in two separate counterbleeding interactions and one double counterbleeding interaction.

(8) Counterbleeding interactions

UR	/mi:k-hin/	/cu:m-it/	/cu:m-hin/
RH	-	cu:mut	cu:mhun
L	me:khin	co:mut	co:mhun
CSS	mekhin	-	comhun
SR	[mekhin]	[co:mut]	[comhun]
Gloss	*'swallow'*	*'destroyed'*	*'destroys'*

- When lowering counterbleeds rounding harmony, the appropriate contextual faithfulness constraint is ID(αhi)/_[αhi], which demands faithfulness in height when the following segment on the vowel tier agrees in height.
- When closed syllable shortening counterbleeds lowering, the relevant contextual faithfulness constraint is ID(long)/[hi], which demands faithfulness to segments which are [hi] in the input, preventing vowel shortening from applying until the vowel is no longer [hi] and lowering has applied.

(9) Constraints for Yawelmani double counterbleeding

ID(long)/[hi]: Assign one * if the value of [long] is changed on a segment which is [high] in the input.

ID(αhi)/_[αhi]: Assign one * if the value of [high] is changed on a segment which precedes a segment with an identical specification for [high] on the vowel tier in the input.

ROUND: Assign one * if the suffix vowel and the preceding root vowel are high and do not match in rounding.[1]

*[long][hi]: Assign one * for every segment which is both [long] and [hi].

*[$\mu\mu\mu$]$_\sigma$: Assign one * for every syllable containing three moras.

ID(long): Assign one * if [long] is added or removed from a segment.

ID(hi): Assign one * if [hi] is added or removed from a segment.

ID(rd): Assign one * if [round] is added or removed from a segment.

[1] This is a simplified version of several constraints that capture the rounding harmony pattern. We adopt this here for explanatory simplicity to focus on the contextual faithfulness constraints and the counterbleeding interaction. The analysis of counterbleeding works in the same way regardless of which constraints are used for the rounding harmony.

(10) Double counterbleeding

Step 1: rounding harmony

/cu:m-hin/	ID(long)/[hi]	*[μμμ]$_\sigma$	ID(long)	ID(αhi)/_[αhi]	*[long][hi]	ID(hi)	ROUND	ID(rd)
→1. cu:mhun		*			*			*
2. cu:mhin		*			*		*W	L
3. co:mhin		*		*W	L	*W		L
4. cumhin	*W	L	*W				*W	L

Step 2: lowering

cu:mhun	ID(long)/[hi]	*[μμμ]$_\sigma$	ID(long)	*[long][hi]	ID(hi)
→ 1. co:mhun		*			*
2. cu:mhun		*		*W	L
3. cumhun	*W	L	*W		

Step 3: closed syllable shortening

co:mhun	*[μμμ]$_\sigma$	ID(long)
→ 1. comhun		*
2. co:mhun	*W	L

4 Counterfeeding Opacity

- Counterfeeding opacity results in surface underapplication (Kiparsky, 2000).
- A rule seems to have not applied on the surface, even though the context for its application is present.

(11) **Counterfeeding in Basque:** Low vowels become mid before vowels, mid become high, low do not become high (Hualde and de Urbina, 2003).

a. /alab<u>**a**</u>-a/ → [alab<u>**e**</u>-a] (↛ *alabi-a)
b. /sem<u>**e**</u>-e/ → [sem<u>**i**</u>-e]

(12) In rules:

a. **Mid to high raising:** e → i / _V
b. **Low to mid raising:** a → e / _V

4.1 Analyzing Counterfeeding with FAITH-UO (Hauser et al., 2014)

- **FAITH-UO:** a set of constraints demanding faithfulness between UR and output.

(13) **ID-UO(*F*)/[α*G*]**
Do not change the value of *F* for segments that are [α*G*] in the UR.

(14) **ID-UO(*F*)/_[α*G*]**
Do not change the value of *F* for segments that are **in the environment of** [α*G*] in the UR.

- **Example:** For Basque,

(15) **ID-UO(hi)/[+low]:** Do not change the value of [α hi] for segments that are [+low] in the UR.

- Referring to the UR at every step of the derivation captures the idea that speakers have access to the lexicon throughout the stages of a phonological derivation.

4.2 Analysis: Basque chain shift

- **Basque: Low becomes mid, mid doesn't become high:**
  /alab$\underline{\text{a}}$-a/ → [alab$\underline{\text{e}}$-a] (↛ *alabi-a)

(16) **Step 1:** Low raises to mid

	/alaba-a/	ID-UO(hi)/[+low]	*low/_V	*mid/_V	ID-IO(hi)
→ 1.	alabe-a				*
2.	alaba-a		* W		L

(17) **Step 2:** Derived mid doesn't raise to high

	/alaba-a/ alabe-a	ID-UO(hi)/[+low]	*low/_V	*mid/_V	ID-IO(hi)
→ 1.	alabe-a			*	
2.	alabi-a	* W		L	

4.3 Analysis: Multi-step chain shift in Nzebi

- FAITH$_{UO}$ constraints can also analyze cases of layered underapplication opacity. We illustrate this with a multi-step chain shift.

- Multiple FAITH$_{UO}$ constraints are needed, one for each individual interaction. As with the layered counterbleeding, when there is an eligible input, both constraints are active.

(18) **Nzebi:** low vowels raise to mid lax, mid lax vowels raise to mid tense, mid tense vowels raise to high (Kirchner, 1996)

/sal/ → [sɛl] (↛ *sel ↛ *sil)
/bɛd/ → [bed] (↛ *bid)
/bet/ → [bit]

(19) **Step 1:** Low raises to mid lax

	/sal/	ID$_{UO}$(ATR)/[$\overline{\text{+low}}$]	ID$_{UO}$(hi)/[$\overline{\text{-ATR}}$]	RAISE	ID$_{IO}$(hi)	ID$_{IO}$(ATR)
→ 1.	sɛl			**		
2.	sal			***W		
3.	sel	*W		*L		*W

(20) **Step 2:** Mid lax doesn't raise to mid tense

	/sal/ sɛl	ID_{UO}(ATR)/[$\overline{\text{+low}}$]	ID_{UO}(hi)/[$\overline{\text{-ATR}}$]	RAISE	ID_{IO}(hi)	ID_{IO}(ATR)
→ 1.	sɛl			**		
2.	sel	*W		*L		*W
3.	sil	*W	*W	L	*W	*

5 Implications

- **Counterbleeding:**
  - Contextual faithfulness constraints have the potential to add unwanted patterns to the typology.
  - We avoid this problem by proposing an algorithm for inducing contextual faithfulness and FAITH-UO constraints as needed on a language-specific basis.
- **Counterfeeding:**
  - Because FAITH-UO specifies a property of the UR, we predict that counterfeeding interactions should only operate over contrastive features.
  - This prediction is observed, to our knowledge.

6 Typology and constraint induction

- These constraints are highly specific and include a lot of information: a specified focus, a specified context, and reference to either the most recent input or the UR.
- To avoid typological problems, we argue these constraints are induced on a language specific basis.

6.1 Counterbleeding

- When free re-ranking is allowed, a pathological prediction emerges from the use of contextual faithfulness constraints (Grammar 4).
- Grammar 4 is the only grammar added to the typology (other than the desired counterbleeding) when the contextual faithfulness constraint is added.
- This "protective" pathology is a result of the specific faithfulness demanded after unpalatalized consonants and occurs when the constraint is undominated.
- With the constraint induction algorithm in place, this language is unlearnable.

(21) HS typology with Bedouin Arabic

	/ʃaribat/	/haːkim/	/haːkimin/	Ranking	Description
1	ʃaribat	haːkim	haːkimin	Max, Id[bk], Max(i)/k_CV>> *ki, *iCV	faithful
2	ʃaribat	haːkʲim	haːkʲimin	*ki, Max, Max(i)/k_CV>> *iCV, Id[bk]	palatalizing
3	ʃaribat	haːkm	haːkmin	*ki, Id[bk]>> Max, Max(i)/k_CV>> *iCV	too many repairs
4	ʃarbat	haːkim	haːkimin	Id[bk], Max(i)/k_CV>> *ki, *iCV>> Max	protective pathology
5	ʃarbat	haːkim	haːkimin	*iCV, Id[bk]>> Max, Max(i)/k_CV>> *ki	delete to repair *iCV
6	ʃarbat	haːkʲim	haːkmin	*ki, *iCV>> Max, Max(i)/k_CV>> Id[bk]	standard HS
7	ʃarbat	haːkʲim	haːkʲmin	*ki, Max(i)/k_CV>> *iCV, Id[bk]>> Max	counterbleeding
8	ʃarbat	haːkm	haːkmin	*ki, *iCV, Id[bk]>> Max, Max(i)/k_CV	too many repairs

- This algorithm is designed to work within a model of learning in an HS framework.
- It is necessary to make many assumptions about the learning process: learner knows candidate set and intended winners.
- When the learner encounters a specific type of ranking problem which we define, the learner attempts to induce a new constraint.
- The pathology is not learnable because the learner doesn't encounter the specified ranking problem given data from the pathological language.
- Detecting inconsistency:
  - A general inconsistency detection procedure such as the one in (Tesar and Smolensky, 1998) will alert the learner that recursive constraint demotion has failed.
  - Learner is trigger to induce contextual faithfulness constraint if:
  - (1) the inconsistency can be located within a single derivation where opposite rankings of a constraint pair are required at two different steps of the derivation, and
  - (2) at the earlier step the current winner satisfies two markedness constraints but the desired winner only satisfies one.
- This procedure will only cause contextual faithfulness constraints to be induced in cases of over-application opacity.

(22) Inconsistency: /ħaːkim-in/→ [ħaːkʲmin]

/ħaːkim-in/	*[ki]	Ident[bk]	*iCV	Max
→ 1. ħaːkʲimin		*	*	
💣 2. ħaːkmin		L	L	*W
3. ħaːkim-in	*W	L	*	

Counterbleeding induction process

Step 1. Identify the faithfulness constraint violated by the current winner in the earlier step of the derivation containing the inconsistency: MAX

Step 2. Align the two markedness constraints which are violated by the faithful candidate but satisfied by the current winner.[2] The segmental overlap becomes the focus of the faithfulness constraint: MAX(i)

Step 3. The total alignment of the two markedness constraints becomes the context for the faithfulness constraint, where the overlapping segment is the focus: MAX(i)/k_CV

- With this procedure in place, all possible enumerations of contextual faithfulness will not exist in universal CON.
- Contextual faithfulness constraints are also not fully re-rankable, preventing the protective pathology from emerging.

6.2 Counterfeeding

- The induction algorithm for counterfeeding is designed to work alongside the algorithm for counterbleeding within the same system.
- Detecting inconsistency:
  - Learner is trigged to induce a constraint under the same conditions, when RCD fails.
  - Unlike counterbleeding, inconsistency cannot be identified in a single derivation.
  - For a single pair of constraints, one ranking is needed at some step in derivation A and the opposite ranking is needed at a later step in derivation B.

(23) **Inconsistency across derivations**

Step 1: Underlying mid raises to high, *[-low, -hi]/_V ≫ IDENT(hi) ("Derivation A")

/seme-e/	*[+low]/_V	*[-low, -hi]/_V	IDENT(hi)	IDENT(low)
→ 1. semi-e			*	
2. seme-e		*W	L	

Step 2: Derived mid doesn't raise to high, IDENT(hi) ≫ *[-low, -hi]/_V ("Derivation B")

alabe-a	*[+low]/_V	*[-low, -hi]/_V	IDENT(hi)	IDENT(low)
→ 1. alabe-a		*		
💣 2. alabi-a		L	*W	

- When conditions for induction are met, the learner attempts to induce a FAITH_{UO} using the algorithm below.

[2] In this example, it is simple to align the markedness constraint as they are written. Depending on how the relevant markedness constraints are defined the process will be more complicated. If the constraints are less segmentally specific, or refer to features instead of segments, the algorithm would involve examining the full definition and segmental extension of the constraints and finding the overlap between the two segmental contexts there.

Counterfeeding induction process

Step 1. Identify the faithfulness constraint violated by the current winner in derivation B: IDENT(hi)

Step 2. Make this faithfulness constraint demand faithfulness between the UR and the current step of the derivation instead of the input and output of the current step: IDENT(hi)$_{UO}$

Step 3. The markedness constraint satisfied by the winner and violated by the faithful candidate in Step 1 of derivation B becomes the context for the faithfulness constraint: IDENT$_{UO}$(hi) / [$\overline{\text{+low}}$]

7 Comparison with other analyses

Contrast Preservation (Łubowicz, 2003)

- Chain shifts analyzed through constraints specifically demanding contrast preservation.
- Sets of derivations (scenarios) evaluated by the constraints.
- Our account: contrast is emergent and does not need to be explicitly demanded by the grammar, faithfulness constraints evaluate inputs and outputs instead of entire derivations and sets of derivations.

OT with Candidate Chains (McCarthy, 2007)

- Candidates consist of chains of output forms from each derivational step.
- EVAL has access to chain and terminal link, PREC constraints evaluate complete derivations.
- Our account: new specific faithfulness constraints do not require changes to EVAL, evaluation is still over inputs and outputs instead of entire derivations.

Serial Markedness Reduction (Jarosz, 2014)

- Introduces a new family of constraints which work within the derivation itself evaluating input-output mappings.
- Adds less technology than OT-CC, constraints evaluate order in which markedness constraints are satisfied.
- Candidates include list of markedness constraints satisfied at each step.
- Our approach: constraints work with standard HS assumptions regarding candidate representation and generation, do not require keeping a record of previous derivational steps or markedness constraints satisfied.

Local Constraint Conjunction (Smolensky, 1995; Ito and Mester, 2003)

- Used as a method of analyzed counterfeeding opacity, but cannot account for other cases of opacity.
- Overpredicts what types of opacity should occur (McCarthy, 1999; Padgett, 2002).
- Our approach: typology constrained through constraint induction only when opaque data presented.

8 Conclusion

- The addition of Contextual Faithfulness and FAITH-UO constraints allows for analysis of counterbleeding and counterfeeding opacity in Harmonic Serialism.
- **Contextual Faithfulness:** Add context of application, which protects the segment/feature until the context is no longer met.
- **FAITH-UO:** Add faithfulness to some property of the Underlying Representation.

(24) **Generalized constraint ranking for opacity:**
Contextual Faithfulness/FAITH-UO >> MARKEDNESS >> FAITH-IO

References

Al-Mozainy, H. Q., 2007. Vowel alternations in a Bedouin Hijazi Arabic dialect: Abstractness and stress. Ph.D. thesis, University of Texas, Austin.

Hauser, I., Hughto, C., Somerday, M., 2014. Faith-UO: Analyzing opacity in Harmonic Serialism. In: Proceedings of the Annual Meetings on Phonology, vol. 1.

Hualde, J. I., de Urbina, J. O., 2003. A grammar of Basque, vol. 26. Walter de Gruyter.

Ito, J., Mester, A., 2003. On the sources of opacity in OT: Coda processes in German. The Syllable in Optimality Theory , 271–303.

Jarosz, G., 2014. Serial markedness reduction. In: Proceedings of the Annual Meetings on Phonology, vol. 1.

Kiparsky, P., 2000. Opacity and cyclicity. The linguistic review 17, 351–366.

Kirchner, R., 1996. Synchronic chain shifts in optimality theory. Linguistic Inquiry , 341–350.

Łubowicz, A., 2003. Contrast preservation in phonological mappings. Ph.D. thesis, University of Massachusetts Amherst.

McCarthy, J. J., 1999. Sympathy and phonological opacity. Phonology 16, 331–399.

McCarthy, J. J., 2007. Hidden generalizations: phonological opacity in Optimality Theory. Equinox.

McCarthy, J. J., 2010. An introduction to harmonic serialism. Language and Linguistics Compass 4, 1001–1018.

Newman, S. S., 1944. Yokuts language of California. 2, Viking Fund.

Padgett, J., 2002. Constraint conjunction versus grounded constraint subhierarchies in optimality theory. University of California, Santa Cruz .

Smolensky, P., 1995. On the structure of the constraint component Con of UG. ROA-86 .

Tesar, B., Smolensky, P., 1998. Learnability in optimality theory. Linguistic Inquiry 29, 229–268.