Giorgio Magri (IJN, ENS)
magrigrg@gmail.com

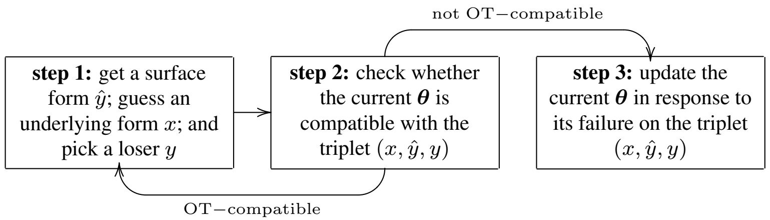
EALing 2010, ENS
Paris, September 10, 2010

The OT online model of the acquisition of phonotactics
Class 4: correctness

Summary — This class introduces the final version of the OT online model of the early stage of the acquisition of phonotactics; shows that the problem of the acquisition of phonotactics in OT cannot be solved in its full generality (it is NP-complete), namely without restrictions on the typologies (i.e. on the constraint set and the generating function); introduces a family of typologies for which the OT online model is provably correct, no matter how the data are sampled.

1 The OT online model of the acquisition of phonotactics

■ **General shape of the model.** Recall from class 1, that an OT online algorithm maintains a current ranking vector $\boldsymbol{\theta}$, initialized to some $\boldsymbol{\theta}^{\text{init}}$, updated thorough the three steps (1).

(1) 

To complete the definition of the model, I need to specify:

(2) a. the initial ranking vector $\boldsymbol{\theta}^{\text{init}}$;
b. the subroutine that provides the underlying form x in step 1;
c. the subroutine that provides the loser form y in step 1;
d. the update rule to be used in step 3.

I spell out my choices in the rest of this section.

■ **Initial ranking vector.** Fikkert and De Hoop (2009, p. 325) note that:

(3) "The recurrent pattern in child language data is that children's output is considerably less marked than the corresponding adult target forms, for segmental, syllabic and higher prosodic structure.

Standardly, (3) is captured by (4); see Smolensky (1996a,b) for theoretical arguments; Jusczyk et al. (2002) for empirical evidence; and Davidson et al. (2004) for a review. The only exception to (4) I know of is Hale and Reiss (1998).

(4) Markedness constraints are initially ranked above faithfulness constraints.

Smith (2000) and Revithiadou and Tzakosta (2004) refine (4) as follows:

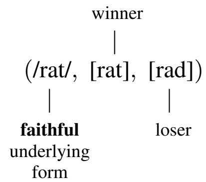
(5) Positional faithfulness constraints are initially ranked above the corresponding general faithfulness constraints.

A ranking vector that satisfies (4) and (5) is *restrictive*, as it corresponds to a smallest language in the typology. I assume that the OT online model starts from a restrictive initial ranking vector.

■ **Choice of the underlying form.** We want to model the early stage of the acquisition of phonotactics. That is the stage prior to morphological awareness. Thus, the child has no access to alternations throughout this stage. As discussed in class 1, it thus makes sense to assume fully faithful underlying forms.

(6) Given a surface form $\hat{y}$, the model assumes a fully faithful underlying form $x = \hat{y}$ in step 1.

Given a language L in the typology, consider all sets of triplets whose winner is a form of L and whose underlying form is identical to the winner form:

(7) 

Consider the comparative rows corresponding to these many triplets and stack them one on top of the other into the *phonotactics comparative tableau* $\mathbf{A} = \mathbf{A}(L)$ corresponding to L. Of course, it has the shape in (8): the faithfulness constraints cannot contain any L.

$$\begin{array}{cc} & \text{faithfulness constraints} \qquad \text{markedness constraints} \\ \text{(8)}\quad \mathbf{A}(L) = & \left[\begin{array}{c|c} \quad\text{E, W}\quad & \quad\text{L, E, W}\quad \end{array}\right] \end{array}$$

In class 1, we commented a bit on assumption (6):

(9) a. it only makes sense provided that the set of underlying forms $\mathcal{X}$ coincides with the set of surface forms $\mathcal{Y}$, so that (6) does not apply to cases such as syllabification, stress assignment, etceteras.
b. Tesar (2008) proves (6) is sound, as the corresponding comparative tableau (8) is OT-compatible (under mild assumptions on the constraint set).

■ **Choice of the loser form.** The most simple-minded strategy is as follows:

(10) Given an underlying/winner form x, the model picks at random a loser $y \in Gen(x)$ in step 1.

Let $\mathbf{A}_x$ be the block of rows of the phonotactics tableau corresponding to triplets whose underlying/winner form is x. In comparative notation, (10) thus becomes:

(11) Given an underlying/winner form x, the model picks at random a row of the corresponding comparative block $\mathbf{A}_x$ in step 1.

This strategy can be refined as follows. Given a tableau $\mathbf{A}$, there exists sometimes a block $\mathbf{A}'$ that captures all the relevant information carried by $\mathbf{A}$, so that the remaining rows $\mathbf{A} \setminus \mathbf{A}'$ are redundant. We can thus use $\mathbf{A}'$ instead of $\mathbf{A}$ for all intents and purposes.

(12) $$\mathbf{A} = \left[\begin{array}{c} \mathbf{A}' \\ \hline \mathbf{A} \setminus \mathbf{A}' \end{array}\right] \begin{array}{l} \longleftarrow \text{crucial information} \\ \\ \longleftarrow \text{redundant information} \end{array}$$

To formalize this idea, let's say that two comparative tableaux $\mathbf{A}$ and $\mathbf{A}'$ (with the same number of columns but possibly a different number of rows) are *OT-equivalent* iff:

(13) A ranking is OT-compatible with $\mathbf{A}$ iff it is OT-compatible with $\mathbf{A}'$.

A sub-block $\mathbf{A}'$ of $\mathbf{A}$ contains all the crucial information carried by $\mathbf{A}$ iff $\mathbf{A}'$ is OT-equivalent with $\mathbf{A}$. Let:

(14) $min(\mathbf{A})$ = a sub-block of $\mathbf{A}$ OT-equivalent to $\mathbf{A}$ that does not admit proper sub-blocks OT-equivalent to $\mathbf{A}$.

The strategy (11) can thus be refined as follows:

(15) Given an underlying/winner form x, the model picks at random a row from $min(\mathbf{A}_x)$ in step 1.

Assumption (15) raises subtle algorithmic issues (is the minimal block unique? can it be computed efficiently?); yet, (15) will play a crucial role in the analysis of the model.

- **Update rule.** In class 1, we concluded that:

(16) The OT online model of the early stage cannot work with demotion only. Rather, we need constraint promotion in order to re-rank the faithfulness constraints too, despite the fact that they are always winner-preferrers.

In classes 2 and 3, we thus worked hard to construct proper promotion/demotion update rules. I will thus use the cautious promotion/demotion update rules deviced:

(17) a. promote every WPC by the total number of undominated LPCs;
b. demote every undominated LPC by the total number of WPCs.

2 Correctness

- **Converging sequences.** Consider a language L in a typology τ; let:

(18) $\mathbf{A} = \mathbf{A}(L)$ be the corresponding phonotactics comparative tableau.

Consider a run of the OT online algorithm on this language L. To start, we have a restrictive initial ranking vector $\boldsymbol{\theta}_{\text{init}}$, that comes with its OT-language L_{init}

(19) $$\begin{tikzcd} \boldsymbol{\theta}_{\text{init}} \arrow[d, dotted, no head] \\ L_{\text{init}} \end{tikzcd}$$

At time 1, we get a row $\mathbf{a}_1$ sampled from $\mathbf{A}(L)$ and update according to (17) to a (possibly different) ranking vector $\boldsymbol{\theta}_1$, that comes with its OT-language L_1.

(20) $$\begin{tikzcd} & \mathbf{a}_1 \arrow[d] \\ \boldsymbol{\theta}_{\text{init}} \arrow[r] \arrow[d, dotted, no head] & \boldsymbol{\theta}_1 \arrow[d, dotted, no head] \\ L_{\text{init}} & L_1 \end{tikzcd}$$

At time 2, we get a row $\mathbf{a}_2$ sampled from $\mathbf{A}(L)$ and update according to (17) to a (possibly different) ranking vector $\boldsymbol{\theta}_2$, that comes with its OT-language L_2.

(21) $$\begin{tikzcd} & \mathbf{a}_1 \arrow[d] & \mathbf{a}_2 \arrow[d] \\ \boldsymbol{\theta}_{\text{init}} \arrow[r] \arrow[d, dotted, no head] & \boldsymbol{\theta}_1 \arrow[r] \arrow[d, dotted, no head] & \boldsymbol{\theta}_2 \arrow[d, dotted, no head] \\ L_{\text{init}} & L_1 & L_2 \end{tikzcd}$$

Classes 2 and 3 guarantee that we will reach a time T after which we will not be able to sample any row from $\mathbf{A}(L)$ that will trigger any update. The ranking vector $\boldsymbol{\theta}^T$ and its corresponding language L_T are called the *final vector* and *final language* $\boldsymbol{\theta}_{\text{fin}}$ and L_{fin}.

(22) $$\begin{tikzcd} & \mathbf{a}_1 \arrow[d] & \mathbf{a}_2 \arrow[d] & \cdots & \mathbf{a}_T \arrow[d] \\ \boldsymbol{\theta}_{\text{init}} \arrow[r] \arrow[d] & \boldsymbol{\theta}_1 \arrow[r] \arrow[d] & \boldsymbol{\theta}_2 \arrow[r] \arrow[d] & \cdots \arrow[r] & \boldsymbol{\theta}_T = \boldsymbol{\theta}_{\text{fin}} \arrow[d] \\ L_{\text{init}} & L_1 & L_2 & \cdots & L_T = L_{\text{fin}} \end{tikzcd}$$

A finite sequence $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \ldots, \mathbf{a}_T$ of rows of $\mathbf{A}(L)$ is *L-convergent* iff there is no row of $\mathbf{A}(L)$ that will trigger an update if fed to the algorithm at time $T+1$.

■ **Correctness.** The OT online algorithm is called *correct* for a typology τ iff there exists an initial ranking vector $\boldsymbol{\theta}^{\text{init}}$ such that for every language L in the typology τ and for every L-converging sequence $\mathbf{a}^1, \ldots, \mathbf{a}^T$ of rows from the corresponding phonotactics comparative tableau $\mathbf{A}(L)$, the language entertained by the algorithm at any time t is a subset of the target language:

(23) $L_t \not\subseteq L$

In particular, at the final time we will have:

(24) $L_{\text{fin}} \subseteq L$

And by definition of L-converging sequence, (24) entails in turn that:

(25) $L_{\text{fin}} = L$

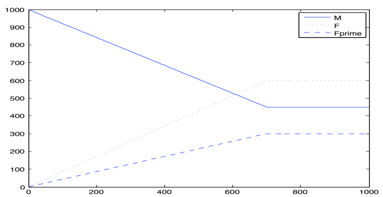
This is an extremely *strong* notion of correctness!

■ **Skeptisism.** Intuitively, we expect that:

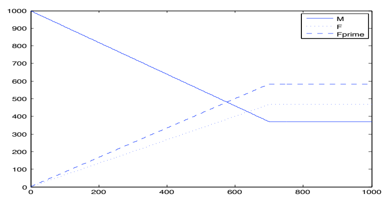
(26) In order for the OT online algorithm with a promotion-demotion update rule to work as a model of the early stage, it is *necessary* that the relative promotion speed of the faithfulness constraints matches their relative target ranking.

Let me illustrate the claim: the target ranking (27a) requires F above F'; the relative promotion speeds of F and F' in (27b) *match* this target ranking: F is promoted faster than F', so that F ends up on top.

(27) a. $F \longrightarrow M \longrightarrow F'$

b. 

In fact, suppose F' was incorrectly promoted faster than F; then F' ends up above F; in order for M to drop below F as required by the target ranking (27a), it also has to drop below F'.

(28) 

But why on earth should we be able to get the right promotion speed of the two faithfulness constraints? and furthermore, why should we get it independently of how we sample the input rows?

■ **Two main claims.** Of course, the OT online model cannot be correct in the general case. But we should not worry about that, in fact we will see in this class that:

(29) Universal correctness cannot be achieved, as the corresponding problem is provably not solvable.

Thus, we need to restrict ourselves to special classes of typologies. And we will se that:

(30) There exist simple yet not trivial class of typologies for which the OT online algorithm is always correct.

Claim (29) is not too surprisingly. Claim (30) is more surprising, in fact:

(31) a. the OT online model is extremely simple, as almost nothing is built into it towards restrictivity, besides the choice of the initial ranking;

b. the task of correctness as stated here is very demanding, as it requires the algorithm to succeed on *any* training sequence.

3 The Ranking problem

■ **Basic formulation.** Suppose that the *actual Gen* function and the *actual* constraint set $\mathcal{C}$ were known. The basic *Ranking problem* would be:

(32) INPUT: a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ that is OT-compatible with $\mathcal{D}$.

But at the current stage of the development of the field, we have no firm knowledge of the actual Gen function and the actual constraint set $\mathcal{C}$.

■ **Strong formulation.** It is thus standard practice to switch to the *universal formulation* (33) of the problem, whereby Gen and $\mathcal{C}on$ figure as arbitrary input to the problem.

(33) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ that is OT-compatible with $\mathcal{D}$.

See Wareham (1998), Eisner (2000), Heinz et al. (2009).

■ **Size of the problem.** Following Tesar (1995) and Tesar and Smolensky (1998), I define the *size* of an instance (33) of the Ranking pbm as in (34):

(34) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ that is OT-compatible with $\mathcal{D}$;
SIZE: $\max\left\{|\mathcal{C}|,\ |\mathcal{D}|,\ |Gen(\mathcal{D})|\right\}$

where $|Gen(\mathcal{D})|$ is the largest number of candidates over all underlying forms in $\mathcal{D}$:

(35) $$|Gen(\mathcal{D})| \stackrel{\text{def}}{=} \max_{(x,\hat{y})\in\mathcal{D}} \Big|Gen(x)\Big|$$

Letting the size of the problem depend on $Gen(\mathcal{D})$ means that a solution algorithm has the time to list all candidates for a given underlying form.

■ **Comparative notation.** A data set $\mathcal{D}$ can be paired up with the corresponding comparative tableau $\mathbf{A}(\mathcal{D})$ in the usual way. The Ranking pbm (34) is thus equivalent to the following pbm:

(36) INPUT: an OT-compatible comparative tableau $\mathbf{A} \in \{\text{L, E, W}\}^{m\times n}$;
OUTPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ that is OT-compatible with $\mathbf{A}$;
SIZE: $\max\Big\{m,\, n\Big\}$.

The equivalence between the original formulation (34) of the Ranking pbm and the reformulation in (36) crucially relies on the two following parallelisms:

(37)			
a.	the size of an instance of the Ranking pbm (34) depends not only on $\lvert\mathcal{C}\rvert$ and $\lvert\mathcal{D}\rvert$, but also on $\lvert Gen(\mathcal{D})\rvert$ in (35).	$\Longleftrightarrow$	the size of an instance of the reformulation (36) depends on the numbers m and n of rows and columns of the tableau.
b.	the universal formulation (34) places no assumptions on the Gen function and on the constraint set $\mathcal{C}$	$\Longleftrightarrow$	the reformulation (36) places no assumptions on the input comparative tableau.

As we saw in class 2, T&S prove that:

■ **Claim 1** The Ranking pbm (34) is tractable. Thus, no harm comes from switching to the universal formulation, at least for the case of the Ranking pbm.

4 The problem of the acquisition of phonotactics

■ **Phonotactics.** To know the phonotactics of a language L means:

(38) a. for every form in L, to know that it is in L;
b. for every form *not* in L, to know that it is *not* in L.

The Ranking pbm (34) captures the positive side (38a) of knowledge of phonotactics but not its negative side (38b).

■ **Problem of the acquisition of phonotactics.** I capture the desideratum (38b) with the minimality condition (39b) on the language corresponding to the output ranking. Thus, (39) is a formalization of the *problem of the Acquisition of Phonotactics* in OT.

(39) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ such that:
a) $\gg$ is OT-compatible with $\mathcal{D}$,
b) there is no ranking $\gg'$ OT-compatible with $\mathcal{D}$ too and such that furthermore $L(\gg') \subsetneqq L(\gg)$.

Cf. Berwick (1985); Manzini and Wexler (1987); Prince and Tesar (2004); Hayes (2004).

■ **Size of the problem.** Let it depend generously on $|\mathcal{X}|$ and $|Gen(\mathcal{X})|$, rather than on $|\mathcal{D}|$ and $|Gen(\mathcal{D})|$ as in the Ranking problem (34).

(40) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking $\gg$ over the constraint set $\mathcal{C}$ such that:
a) $\gg$ is OT-compatible with $\mathcal{D}$,
b) there is no ranking $\gg'$ OT-compatible with $\mathcal{D}$ too and such that furthermore $L(\gg') \subsetneqq L(\gg)$.
SIZE: $\max\Big\{|\mathcal{C}|,\, |\mathcal{X}|,\, |Gen(\mathcal{X})|\Big\}$.

This is a very generous formulation of the problem, in fact:

(41) a. the underlying form is provided for every given winning surface form;
b. the size of an instance of the pbm is as generous as it can be.

Since I want to show intractability, it is good to consider a generous formulation.

5 Prince and Tesar's (2004) reformulation

■ **Strictness measures.** A *strictness measure* is a function μ with the following shape:

(42) $$\mu:\ \gg \longrightarrow \text{a number } \mu(\gg) \ = \ \text{a relative measure of the cardinality of the corresponding language } L(\gg)$$

in such a way that any solution of pbm (43) is a solution of the pbm of the Acquisition of Phonotactics (39).

(43) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking with minimal measure μ among those OT-comp. with $\mathcal{D}$.

■ **A concrete example.** Assume that the constraint set is split up into the subset $\mathcal{F}$ of faithfulness constraints and the subset $\mathcal{M}$ of markedness constraints:

(44) $$\mathcal{C}on = \mathcal{F} \cup \mathcal{M}$$

The function μ_{PT} in (45) maps a ranking $\gg$ to the number $\mu_{\mathrm{PT}}(\gg)$ of pairs of a faithfulness and a markedness constraint such that the former is $\gg$-ranked above the latter.

(45) $\mu_{\mathrm{PT}}(\gg) \overset{\mathrm{def}}{=} \left| \left\{ (F, M) \in \mathcal{F} \times \mathcal{M} \,\middle|\, F \gg M \right\} \right|$

Prince and Tesar (2004) conjecture that the function μ_{PT} in (45) is a strictness measure.

- **Prince &Tesar's reformulation.** Problem (43) with the mesure μ_{PT} in (45) is *Prince and Tesar's reformulation* of the pbm of the Acquisition of Phonotactics:[1]

(46) INPUT: a) the Gen function and the constraint set $\mathcal{C}$,
b) a finite data set $\mathcal{D}$ of underlying/winner form pairs, OT-compatible with some ranking;
OUTPUT: a ranking with minimal measure μ_{PT} among those OT-comp. with $\mathcal{D}$;
SIZE: $\max\{|\mathcal{C}|, |\mathcal{D}|, |Gen(\mathcal{D})|\}$.

Strictness measures determine relative strictness without looking at the entire set $\mathcal{X}$. Thus, the size of (46) depends on $|\mathcal{D}|$ and $|Gen(\mathcal{D})|$, rather than on $|\mathcal{X}|$ and $|Gen(\mathcal{X})|$ as for the original formulation (40) of the pbm of the Acquisition of Phonotactics.

6 The pbm of the acquisition of phonotactics cannot be solved

- **Claim 2** The universal formulation of the pbm of the Acquisition of Phonotactics in OT is NP-complete and thus cannot be solved by any algorithm, both in its original formulation (40) as well as in Prince & Tesar's reformulation (46).
- *Proof.* The *CyclicOrdering* problem as a decision problem is:

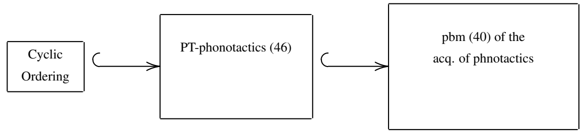
(47) INPUT: a) a finite set A;
b) a set $T \subseteq A \times A \times A$ of triplets of elements of A;
OUTPUT: "yes" iff T is *linearly cyclically compatible*;
SIZE: the cardinality $|A|$ of the given set.

where the set of triplets T is called *linearly cyclically compatible* iff there exist a linear order $>$ on the set A such that one of the following holds for every triplet $(a, b, c) \in T$:

(48) a. $a < b < c$,
b. $b < c < a$,
c. $c < a < b$.

Galil and Megiddo (1977) prove NP-completeness of Cyclic Ordering by reduction from *3-Satisfability*; Cyclic Ordering is problem [MS2] in (Garey and Johnson, 1979, p. 279). I prove claim 2 by constructing the following reductions:

[1] The Ranking pbm (33) corresponds to *Empirical Risk Minimization* in the Statistical Learning literature, while pbm (46) corresponds to a *regularized* version thereof, with regularization function μ.

(49) 

In other words I show that:

(50) a. Each instance of the Cyclic Ordering pbm can be mapped into an instance of the PT-Phonotactics pbm such that the answer to the former is "yes" iff the answer to the latter is 'yes";
b. this mapping says that, if the PT-Phonotactics pbm were solvable, then Cyclic Ordering would be solvable too;
c. since Cyclic Ordering is not solvable, then I conclude that PT-Phonotactics is not solvable either.

and analogously for the embedding of PT-phonotactics into the original pbm (40) of the acquisition of phonotactics. □

- **Remarks.** The proof actually shows that the pbm of the acquisition of Phonotactics (40) remains NP-hard even when:

(51) a. the data have the simplest "disjunctive structure", namely for each underlying/winner/loser form there are at most two winner-preferring constrains;[2]
b. the data have the property that the faithfulness constraints are never loser-preferring (as in the case of phonotactics comparative tableaux).

Claim 2 says that the switch to the universal formulation is harmless for the easy Ranking pbm, as shown by claim 1, but harmful for the pbm of the Acquisition of Phonotactics.

- **Interim conclusion.** We have seen two properties of the OT online algorithm, together with the corresponding computational problems. The correspondence consists in the fact that, if the property holds, then the OT online model "solves" the corresponding problem.

(52) a. *convergence* for the OT online algorithm ——— Ranking problem
b. *correctness* for the OT online algorithm ——— Problem of the acquisition of phonotactics

With respect to (52a):

(53) a. We have seen that the Ranking problem is "easy", in the sense that the intrinsic logic of OT provides by itself enough structure to solve the problem;
b. thus, we can stick with the universal formulation of the problem, without introducing further structure by restricting to special subclasses of typologies;

[2] Of course, if there were a unique winner-preferring constraint per underlying/winner/loser form triplet, then the data would be OT-compatible with a unique ranking, and thus the pbm of the Acquisition of Phonotactics (46) would reduce to the Ranking pbm (33).

c. in classes 2 and 3 we have thus demanded that the corresponding property of convergence holds universally for the OT online algorithm, namely that the algorithm converges for any typology.

With respect to (52b):

(54) a. We have just sees that the problem of the acquisition of Phonotactics is "hard", in the sense that the intrinsic logic of OT does not provide by itself enough structure to solve the problem;

b. thus, we cannot stick with the universal formulation of the pbm, rather need to introduce further structure by restricting to special classes of typologies;

c. thus, we cannot demand that the corresponding property of correctness holds universally for the OT online algorithm, namely it is correct for any typology.

This leads to the following research question:

(55) Are there families of typologies for which we can ensure correctness for the OT online algorithm?

This is one more instance of the modeling strategy (56) of *Cognitive Computational Phonology*, outlined in class 1.

(56) a. Single out the building blocks of the learning task (e.g. the Ranking problem, the problem of the Acquisition of Phonotactics, etceteras);

b. study their complexity (e.g. claims 1 and 2 above);

c. devise algorithms that solve these problems up to their complexity class.

In the rest of this class, I concentrate on research question (55).

7 Some case studies

■ **Korean typology.** The universal specifications are as follows, from Hayes (2004):

(57) a.
$$\begin{array}{c}\text{set of}\\ \text{underlying}\\ \text{forms}\end{array} = \begin{array}{c}\text{set of}\\ \text{surface}\\ \text{forms}\end{array} = \left\{\begin{array}{cccc} \text{ta,} & \text{da,} & \text{t}^h\text{a,} & \text{d}^h\text{a,}\\ \text{at,} & \text{ad,} & \text{at}^h\text{,} & \text{ad}^h\text{,}\\ \text{ata,} & \text{ada,} & \text{at}^h\text{a,} & \text{ad}^h\text{a} \end{array}\right\}$$

b.
$$\begin{array}{c}\text{set of}\\ \text{constraints}\end{array} = \left\{\begin{array}{l} F_1 = \text{IDENT[VOICE]}\\ F_2 = \text{IDENT[ASPIRATION]}\\ F_3 = \text{IDENT[VOICE]/ONSET}\\ F_4 = \text{IDENT[ASPIRATION]/ONSET}\\ M_1 = {}^*\text{[-SONORANT, +VOICE]}\\ M_2 = {}^*\text{[+VOICE][-VOICE][+VOICE]}\\ M_3 = {}^*\text{[+SPREAD GLOTTIS]}\\ M_4 = {}^*\text{[+SPREAD GLOTTIS, +VOICE]} \end{array}\right\}$$

c. Gen is an equivalence relation defined by the following classes:

i. ta $\overset{\text{Gen}}{\sim}$ da $\overset{\text{Gen}}{\sim}$ t^{h}a $\overset{\text{Gen}}{\sim}$ d^{h}a

ii. at $\overset{\text{Gen}}{\sim}$ ad $\overset{\text{Gen}}{\sim}$ at^h $\overset{\text{Gen}}{\sim}$ ad^h

iii. ada $\overset{\text{Gen}}{\sim}$ ata $\overset{\text{Gen}}{\sim}$ ad^{h}a $\overset{\text{Gen}}{\sim}$ at^{h}a

The corresponding typology contains 40 languages. I got that:

(58) The OT online algorithm converges to the correct ranking for all 40 languages (with the rows fed in step 1 sampled uniformly from the input tableau).

■ **Azba typology.** The universal specifications are as follows, from Prince and Tesar (2004):

(59) a.
$$\begin{array}{c}\text{set of}\\ \text{underlying}\\ \text{forms}\end{array} = \begin{array}{c}\text{set of}\\ \text{surface}\\ \text{forms}\end{array} = \left\{\begin{array}{cccc} \text{pa,} & \text{ba,} & \text{ap,} & \text{ab,}\\ \text{sa,} & \text{za,} & \text{as,} & \text{az,}\\ \text{apsa,} & \text{apza,} & \text{absa,} & \text{abza,}\\ \text{aspa,} & \text{azpa,} & \text{asba,} & \text{azba} \end{array}\right\}$$

b.
$$\begin{array}{c}\text{set of}\\ \text{constraints}\end{array} = \left\{\begin{array}{lcl} F_1 & = & \text{IDENT[STOP-VOICING]}\\ F_2 & = & \text{IDENT[FRICATIVE-VOICING]}\\ F_3 & = & \text{IDENT[STOP-VOICING]/ONSET}\\ F_4 & = & \text{IDENT[FRICATIVE-VOICING]/ONSET}\\ M_1 & = & \text{AGREE[VOICE]}\\ M_2 & = & {}^*\text{[+STOP-VOICING]}\\ M_3 & = & {}^*\text{[+FRICATIVE-VOICING]} \end{array}\right\}$$

c. Gen is an equivalence relation defined by the following classes:

i. pa $\overset{\text{Gen}}{\sim}$ ba

ii. ap $\overset{\text{Gen}}{\sim}$ ab

iii. sa $\overset{\text{Gen}}{\sim}$ za

iv. as $\overset{\text{Gen}}{\sim}$ az

v. apsa $\overset{\text{Gen}}{\sim}$ apza $\overset{\text{Gen}}{\sim}$ absa $\overset{\text{Gen}}{\sim}$ abza

vi. aspa $\overset{\text{Gen}}{\sim}$ azpa $\overset{\text{Gen}}{\sim}$ asba $\overset{\text{Gen}}{\sim}$ azba

The corresponding typology contains 37 languages. By the symmetry between [STOP-VOICING] and [FRICATIVE-VOICING], I need to consider only 19 of them. I got that:

(60) The OT online algorithm converges to the correct ranking in the case of 18 out of these 19 languages (with the rows fed in step 1 sampled uniformly from the input tableau).

The language where the model returns a superset language is:

(61)
$$\left\{\begin{array}{cccc} \text{pa,} & \text{ba,} & \text{ap,} & \text{ab,}\\ \text{sa,} & & \text{as,} & \\ \text{apsa,} & & & \text{abza,}\\ \text{aspa,} & & & \text{azba} \end{array}\right\}$$

■ **Vowel typology.** Given the following feature matrix:

(62)

	y	i	o	e	A	a
HIGH	+	+	-	-	-	-
LOW	-	-	-	-	+	+
ROUND	+	-	+	-	+	-

Consider the following universal specifications:

(63) a. $$\text{set of underlying forms} = \text{set of surface forms} = \left\{\begin{array}{ccc} \#y & \text{-y-} & y\# \\ \#i & \text{-i-} & i\# \\ \#o & \text{-o-} & o\# \\ \#e & \text{-e-} & e\# \\ \#A & \text{-A-} & A\# \\ \#a & \text{-a-} & a\# \end{array}\right\}$$

b. $$\text{set of constraints} = \left\{\begin{array}{ll} F_1 = \text{IDENT[HIGH]} & M_1 = *[\text{-HIGH, -LOW}] = \{o, e\} \\ F_2 = \text{IDENT[LOW]} & M_2 = *[\text{WORD-FINAL HIGH}] \\ F_3 = \text{IDENT[ROUND]} & M_3 = *[\text{NON-INITIAL, ROUND}] \end{array}\right\}$$

c. *Gen* is an equivalence relation defined by the following classes:

i. $\#y \overset{Gen}{\sim} \#i \overset{Gen}{\sim} \#o \overset{Gen}{\sim} \#e \overset{Gen}{\sim} \#A \overset{Gen}{\sim} \#a$

ii. $\text{-y-} \overset{Gen}{\sim} \text{-i-} \overset{Gen}{\sim} \text{-o-} \overset{Gen}{\sim} \text{-e-} \overset{Gen}{\sim} \text{-A-} \overset{Gen}{\sim} \text{-a-}$

iii. $\#y \overset{Gen}{\sim} i\# \overset{Gen}{\sim} o\# \overset{Gen}{\sim} e\# \overset{Gen}{\sim} A\# \overset{Gen}{\sim} a\#$

The corresponding typology contains 10 languages. I got that:

(64) The OT online algorithm converges to the correct ranking for all 10 languages (with the rows fed in step 1 sampled uniformly from the input tableau).

8 A framework for typologies

- **Universal specifications.** Recall from class 1, that a 4-tuple $\tau = (\mathcal{X}, \mathcal{Y}, Gen, \mathcal{C})$ as in (65) is called the *universal specifications* of a typology.

(65)
$\mathcal{X}$: (finite) set of *underlying forms*;
$\mathcal{Y}$: (finite) set of *surface forms*;
$Gen(x) \subseteq \mathcal{Y}$: set of *candidate* surface forms for the underlying form x;
$\mathcal{C} = \{C_1, \ldots, C_n\}$: set of n *constraints*.
Constraint C_k takes a pair (x, y) of an underlying form $x \in \mathcal{X}$ and a corresponding candidate $y \in Gen(x)$ and returns a nonnegative number $C_k(x, y)$, called the *number of violations*.

- **Set of forms.** Assume there are N partial binary features (i.e. take value 0 or 1, or $\#$ if undefined). Phonology is feature-based (i.e. a form is characterized by its feature values). Thus, the set of forms is defined as follows:

(66) $\mathcal{X}$ = a set of N-tuples $\mathbf{x} = (x_1, \ldots, x_i, \ldots, x_N)$ of 0's, 1's and $\#$'s.

Wlg, I assume that:

(67) For each feature, the marked value is 1, while 0 is the unmarked value.

For example, in the case of the Azba typology, we have:

(68) a. feature 1 = STOP-VOICING;
feature 2 = FRICATIVE VOICING

b. $$\left\{\begin{array}{cccc} \text{pa} = (0, \#), & \text{ba} = (1, \#), & \text{sa} = (\#, 0), & \text{za} = (\#, 1), \\ \text{apsa} = (0, 0), & \text{apza} = (0, 1), & \text{absa} = (1, 0), & \text{abza} = (1, 1) \end{array}\right\}$$

The set of candidate surface forms is identical to the set of surface forms:

(69) $\mathcal{Y} = \mathcal{X}$

- **Generating function.** The *Gen* function takes a form $\mathbf{x}$ and returns the set $Gen(\mathbf{x})$ of all forms in $\mathcal{Y} = \mathcal{X}$ that can be obtained from $\mathbf{x}$ by changing in all possible ways the values of the features that $\mathbf{x}$ is defined for. More explicitly:

(70) $$Gen(\mathbf{x}) = \left\{\mathbf{y} \in \mathcal{X} \,\middle|\, \mathbf{y} \text{ is defined for the same features } \mathbf{x} \text{ is}\right\}$$

For example, in the case of the Azba typology, we have:

(71) $$Gen(\text{/apsa/}) = Gen((0,0)) = \left\{\begin{array}{c} [\text{apsa}] = (0,0), \\ [\text{apza}] = (0,1), \\ [\text{absa}] = (1,0), \\ [\text{abza}] = (1,1) \end{array}\right\}$$

- **Constraint set.** The constraint set can contain three types of constraints:

(72) $$\mathcal{C} \subseteq \left\{\begin{array}{lcl|l} F_i & = & \textit{faithfulness constraint} \text{ corresponding to the feature } i & \\ M_i & = & \textit{simple markedness constraint} \text{ corresponding to the feature } i & i, j = 1, \ldots, N \\ M_{i,j} & = & \textit{binary markedness constraint} \text{ corresponding to the two (different) features } i \text{ and } j & \end{array}\right\}$$

The *faithfulness constraint* F_i corresponding to feature i is defined as follows for every underlying form $\mathbf{x} = (x_1, \ldots, x_N) \in \mathcal{X}$ and every corresponding candidate form $\mathbf{y} = (y_1, \ldots, y_N) \in Gen(\mathbf{x})$.

(73) $$F_i(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{if both } \mathbf{x} \text{ and } \mathbf{y} \text{ are defined for feature } i \text{ and } x_i \neq y_i \\ 0 & \text{otherwise} \end{cases}$$

For example in the case of the Azba typology with the positions in (68a):

(74) IDENT[STOP-VOICING] = F_1
IDENT[FRICATIVE-VOICING] = F_2

The *simple markedness constraint* M_i corresponding to feature i is defined as follows for every form $\mathbf{x} = (x_1, \ldots, x_N) \in \mathcal{X}$:

(75) $M_i(\mathbf{x}) = \begin{cases} 1 & \text{if } x_i = 1 \\ 0 & \text{otherwise} \end{cases}$

For example in the case of the Azba typology with the positions in (68a):

(76) *[STOP-VOICING] $= M_1$
*[FRICATIVE-VOICING] $= M_2$

The *binary markedness constraint* $M_{i,j}$ corresponding to features i and j (with $i \neq j$) is defined as follows for every form $\mathbf{x} = (x_1, \ldots, x_N) \in \mathcal{X}$; the set μ of feature combinations punished by $M_{i,j}$ is called its *markedness pattern*.

(77) $M_{i,j}(\mathbf{x}) = \begin{cases} 1 & \text{if } (x_i, x_j) \in \mu \\ 0 & \text{otherwise} \end{cases}$
where μ is some subset of $\{0,1\}^2$.

For example in the case of the Azba typology with the positions in (68a):

(78) AGREE[VOICING] = $M_{1,2}$ with $\mu = \{(0,1), (1,0)\}$

and in the case of the Korean typology with features 1 and 2 being VOICING and ASPIRATION:

(79) *[+SPREAD-GLOTTIS, +VOICE] $= M_{1,2}$ with $\mu = \{(1,1)\}$

Binary markedness constraints are responsible for feature interaction.

■ **Remarks.**

(80) a. The faithfulness constraints $F_1, \ldots, F_N$ are Ident-type faithfulness constraints; there are no DEP and MAX in this framework.
b. I assume that for each pair of features, there can be at most one binary markedness constraint that targets those features

■ **Preview of the main claim.** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77) corresponding to N features. Assume that:

(81) a. *The mode of feature interaction is "simple":*
no binary markedness constraint punishes a form that is unmarked with respect to both features targeted, namely there is no markedness pattern μ that contains (0,0).
b. *The amount of feature interaction is "limited":*
each feature interacts with at most another feature, namely there are no two binary markedness constraints $M_{4,7}$ and $M_{7,2}$ that both target feature 7.
c. *The set of candidates is "rich enough":*
if two features interact, then the set of forms contains all four possible combinations of those features.

Then, I will prove that the OT online algorithm is correct for the typology τ. And we will see applications to the Azba typology and the vowel typology.

9 First part: simple interaction and rich candidate sets

■ **Simple mode of feature interaction.** In this section, we want to understand what it means that the mode of interaction between two features is simple. Thus, consider the case of the typology (66)-(77) with $N = 2$ features. Here is the idea:

(82) The mode of feature interaction enforced by a binary markedness constraint $M_{1,2}$ is simple iff $M_{1,2}$ interacts "smoothly" with the rest of the constraint set, in the sense that$M_{1,2}$ does not punish a form which is unmarked with respect to both simple markedness constraints M_1 and M_2.

As I have assumed that the markedness value of any feature is 1, then (82) says that the mode of interaction between the two features is simple provided that the corresponding binary markedness constraint does not punish (0, 0).

■ **Rich sets of candidates.** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77) with N features. The Gen function is *complete* for a form $\mathbf{x}$ provided that $Gen(\mathbf{x})$ contains "all possible" candidates:

(83) $|Gen(\mathbf{x})| = 2^{\#\text{ of features } \mathbf{x} \text{ is defined for}}$

For instance, consider $\mathbf{x} = (1, 0, \#)$; Gen in (a) is complete while that in (b) is not.

(84) a. $Gen(\mathbf{x}) = \left\{ \begin{array}{c} (1,0,\#) \\ (1,1,\#) \\ (0,0,\#) \\ (0,1,\#) \end{array} \right\}$ b. $Gen(\mathbf{x}) = \left\{ \begin{array}{c} (1,0,\#) \\ \\ (0,0,\#) \\ (0,1,\#) \end{array} \right\}$

Gen is *complete* iff it is complete for any form.

■ **Claim 3** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77) with only $N = 2$ features. Assume that:

(85) a. The markedness pattern of the unique binary markedness constraint in $\mathcal{C}$ (if any) does not contain [00].
b. The Gen function is complete.

Then, there exists an initial ranking vector such that the OT online algorithm is always correct on the typology τ.

■ **Counterexample against markedness patterns that contain [00].** Consider the following typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$, that fits into the scheme (66)-(77):

(86) a. $\mathcal{X} = \Big\{(00), (01), (10), (11)\Big\}$
b. $Gen(\mathbf{x}) = \mathcal{X}$
c. $\mathcal{C} = \Big\{F_1,\ F_2,\ M_1,\ M_2, M\Big\}$,
where the binary markedness constraint M is only violated by $\mathbf{x} = (0, 0)$.

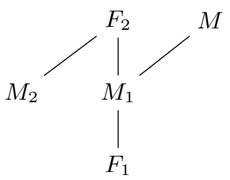
The following language belongs to the typology τ:

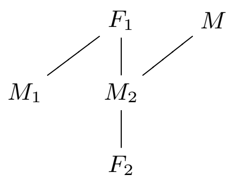
(87) $L = \{(01), (10)\}$

The phonotactics comparative tableau $\mathbf{A}(L)$ corresponding to L is as follows:

$$(88)\quad \mathbf{A}(L) = \begin{array}{r|cc|cc|c|} & F_1 & F_2 & M_1 & M_2 & M \\ \hline (1,0),(1,1) & & \text{W} & & \text{W} & \\ (1,0),(0,0) & \text{W} & & \text{L} & & \text{W} \\ (1,0),(0,1) & \text{W} & \text{W} & \text{L} & \text{W} & \\ \hline (0,1),(1,1) & \text{W} & & \text{W} & & \\ (0,1),(0,0) & & \text{W} & & \text{L} & \text{W} \\ (0,1),(1,0) & \text{W} & \text{W} & \text{W} & \text{L} & \end{array}$$

A ranking $\gg$ generates the language L iff it is a refinement of one of the two following symmetric partial orders:

(89) a. 

b. 

Note that:

(90) a. The first row of both blocks in (88) does not trigger any update (it has no L's and thus is always OT compatible with the current ranking vector);

b. The second row of both blocks in (88) triggers at most one update (it has a W corresponding to a markedness constraint that is never demoted).

Thus, for all intent and purposes, the input tableau consists of only the third row of the two blocks in (88), namely the effective input tableau is as follows:

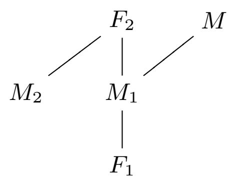
$$(91)\quad \begin{array}{r|cc|cc|c|} & F_1 & F_2 & M_1 & M_2 & M \\ \hline (1,0),(0,1) & \text{W} & \text{W} & \text{L} & \text{W} & \\ \hline (0,1),(1,0) & \text{W} & \text{W} & \text{W} & \text{L} & \end{array}$$

There is no way that the OT online algorithm can start from a ranking vector that assigns the same ranking value to F_1 and F_2 and converge on a ranking vector that represents one of the two rankings in (89) if fed with the two rows in (91).

■ **Partial rankings.** As anticipated in section 2, the though cases are those where the two faithfulness constraints F_1 and F_2 need to be ranked wrt each other. When you look at a typology, there are lots and lots of languages that are simple because F_1 and F_2 don't need to be ranked wrt each other. Since so far we have defined rankings as total orders on the constraint set, we have no way of capturing the notion that F_1 and F_2 need not be ranked wrt each other. Thus:

(92) From now on, a ranking $\gg$ is a *partial* order over the constraint set, rather than a *total* order as assumed so far.

Thus, (93) is a ranking in this new sense, as it is a partial order over the constraint set. This ranking does not rank, say, F_2 and M wrt each other.

(93) 

The partial order (93) admits various refinements into a total order, for instance:

(94) $M \gg F_2 \gg M_1 \gg M_2 \gg F_1$
$F_2 \gg M_2 \gg M_1 \gg M_2 \gg F_1$
$\vdots$

A partial order such as (93) is called *OT-compatible* with a comparative tableau iff each one of its total refinements is OT compatible with that tableau in the usual sense.

■ *Proof.* In principle, claim 3 could be proved by case inspection:

(95) a. list all typologies that satisfy the hypotheses (there are not too many),

b. list all languages for each typology;

c. for each language, study the behavior of the OT online algorithm on the corresponding phonotactics tableaux.

My actual proof is indeed just a refinement of this brute-force strategy.[3] Let's sort the languages in a typology as follows:

(96) language L
- **(a)** is generated by a ranking that does not rank F_1 and F_2 wrt each other
- is only generated by rankings that rank F_1 and F_2 wrt each other, say F_2 above F_1
  - **(b)** in order to rank M_1 in between
  - **(c)** in order to rank M_2 in between
  - **(d)** in order to rank M in between

Thus I need to consider four cases (96a)-(96d). It turns out that:

(97) a. *Case (a)*: the OT online algorithm is trivially correct;

b. *Case (b)*: the OT online algorithm is correct;

[3]The refinement might especially turn out useful for the extension to the case $N = 3$, that I am working on right now.

c. *Case (c)*: is impossible;

d. *Case (d)*: the OT online algorithm is correct.

Claim (97a) is very intuitive. It is not hard to show that in case (97b), the input tableau can only have one of the following shapes and in each case the algorithm converges to the correct final ranking.

(98) a.

$$\begin{array}{cc}
 & \begin{array}{ccccc} F_1 & F_2 & M_1 & M_2 & M \end{array} \\
\begin{array}{c} \\ \Longrightarrow \\ \\ \\ \\ \\ \end{array} & \left[\begin{array}{ccccc} \text{W} & & \text{L} & & \text{W} \\ & \text{W} & & \text{L} & \\ \text{W} & \text{W} & \text{L} & \text{L} & \\ \hline \text{W} & & \text{W} & & \\ & \text{W} & \text{W} & & \text{W} \\ \text{W} & \text{W} & \text{W} & \text{W} & \end{array}\right]
\end{array}$$

b.

$$\begin{array}{cc}
 & \begin{array}{ccccc} F_1 & F_2 & M_1 & M_2 & M \end{array} \\
\begin{array}{c} \Longrightarrow \\ \\ \Longrightarrow \\ \\ \\ \\ \\ \end{array} & \left[\begin{array}{ccccc} & \text{W} & & \text{L} & \\ \hline \text{W} & & \text{L} & & \text{W} \\ & \text{W} & & \text{L} & \\ \text{W} & \text{W} & \text{L} & \text{L} & \\ \hline \text{W} & & \text{W} & & \\ & \text{W} & \text{W} & & \text{W} \\ \text{W} & \text{W} & \text{W} & \text{W} & \end{array}\right]
\end{array}$$

c.

$$\begin{array}{cc}
 & \begin{array}{ccccc} F_1 & F_2 & M_1 & M_2 & M \end{array} \\
\begin{array}{c} \Longrightarrow \\ \\ \\ \\ \\ \\ \\ \end{array} & \left[\begin{array}{ccccc} & \text{W} & & \text{L} & \\ \hline \text{W} & & \text{L} & & \text{W} \\ & \text{W} & & \text{L} & \text{W} \\ \text{W} & \text{W} & \text{L} & \text{L} & \\ \hline \text{W} & & \text{W} & & \text{W} \\ & \text{W} & \text{W} & & \text{W} \\ \text{W} & \text{W} & \text{W} & \text{W} & \end{array}\right]
\end{array}$$

Cases (97c) and (97d) are treated similarly. □

10 Second part: simple feature interactions

■ **Factor typologies.** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77). Split up the feature set $\{1, \ldots, N\}$ into two disjoint sets Φ', Φ''. Wlg, assume that:

(99) $$\{1, \ldots, N\} = \underbrace{\{1, \ldots, M\}}_{\Phi'} \cup \underbrace{\{M+1, \ldots, N\}}_{\Phi''}$$

Split up each form $\mathbf{x} \in \mathcal{X}$ into an M-tuple $\mathbf{x}'$ with the first M feature values of $\mathbf{x}$ and an $(N-M)$-tuple $\mathbf{x}''$ with the remaining $N-M$ feature values.

(100) $$\begin{array}{ccc} & \mathbf{x} = (\overbrace{x_1, \ldots, x_M,}^{\text{first M features}} \mid \overbrace{x_{M+1}, \ldots, x_N}^{\text{remaining N}-\text{M features}}) & \\ \swarrow & & \searrow \\ \mathbf{x}' = (x_1, \ldots, x_M) & & \mathbf{x}'' = (x_{M+1}, \ldots, x_N) \end{array}$$

Collect all the M-tuples $\mathbf{x}'$ thus defined in a set of forms $\mathcal{X}'$ and all the $(N-M)$-tuples $\mathbf{x}''$ thus defined into a set of forms $\mathcal{X}''$, as in (101).

(101) a. $\mathcal{X}' = \Big\{\mathbf{x}' \in \{0, 1, \#\}^M \;\Big|\; \text{there exists } \mathbf{x}'' \text{ s.t. } (\mathbf{x}', \mathbf{x}'') \in \mathcal{X}\Big\}$

b. $\mathcal{X}'' = \Big\{\mathbf{x}' \in \{0, 1, \#\}^{N-M} \;\Big|\; \text{there exists } \mathbf{x}' \text{ s.t. } (\mathbf{x}', \mathbf{x}'') \in \mathcal{X}\Big\}$

Define the generating functions Gen' and Gen'' as in (70), given explicitly in (102).

(102) a. $Gen'(\mathbf{x}') = \Big\{\mathbf{y}' \in \mathcal{X}' \;\Big|\; dom(\mathbf{y}') = dom(\mathbf{x}')\Big\}$

b. $Gen''(\mathbf{x}'') = \Big\{\mathbf{y}'' \in \mathcal{X}'' \;\Big|\; dom(\mathbf{y}'') = dom(\mathbf{x}'')\Big\}$

Consider the two constraint sets $\mathcal{C}'$ and $\mathcal{C}''$ as in (103). Note $\mathcal{C}$ might contain binary markedness constraints that target a feature in Φ' and one in Φ'', and thus do not belong to neither $\mathcal{C}'$ nor $\mathcal{C}$.

(103) a. $\mathcal{C}'$ = set of those constraints in $\mathcal{C}$ that target only features in Φ';

b. $\mathcal{C}''$ = set of those constraints in $\mathcal{C}$ that target only features in Φ''.

The two typologies $\tau' = (\mathcal{X}', Gen', \mathcal{C}')$ and $\tau'' = (\mathcal{X}'', Gen'', \mathcal{C}'')$ are called the two *factor typologogies* corresponding to the partition Φ', Φ'' of the feature set.

■ **Factorization of comparative blocks.** We are interested in the following question:

(104) If I know something about the two factor typologies τ' and τ'', what can I say about the original typology τ?

To make (104) concrete, consider an arbitrary form $\mathbf{x} \in \mathcal{X}$. Split it up into $\mathbf{x} = (\mathbf{x}'\,\mathbf{x}'')$ as in (100). Consider the corresponding comparative blocks:

(105) a. $\mathbf{A}_{\mathbf{x}}$ = comparative block corresponding to $\mathbf{x}$ w.r.t. typology τ;

b. $\mathbf{A}'_{\mathbf{x}'}$ = comparative block corresponding to $\mathbf{x}'$ w.r.t. typology τ';

c. $\mathbf{A}''_{\mathbf{x}''}$ = comparative block corresponding to $\mathbf{x}''$ w.r.t. typology τ''.

Split up the columns of $\mathbf{A}_{\mathbf{x}}$ into three groups:

(106) a. columns corresponding to constraints in $\mathcal{C}'$;

b. columns corresponding to constraints in $\mathcal{C}''$.

c. columns corresponding to constraints in $\mathcal{C}$ but not in $\mathcal{C}'$ nor $\mathcal{C}''$.

Split up the rows of $\mathbf{A}_{\mathbf{x}}$ into three sets:

(107) a. rows corresponding to losers that only differ from $\mathbf{x}$ for features in Φ';
b. rows corresponding to losers that only differ from $\mathbf{x}$ for features in Φ'';
c. rows corresponding to losers that differ from $\mathbf{x}$ for both features in Φ'' and Φ''.

The comparative block $\mathbf{A}_\mathbf{x}$ thus has the shape in (108).

(108)
$$\mathbf{A}_\mathbf{x} = \begin{array}{c} \begin{array}{ccc} \quad\mathcal{C}'\quad & \quad\mathcal{C}''\quad & \mathcal{C}\setminus(\mathcal{C}'\cup\mathcal{C}'') \end{array} \\ \left[\begin{array}{c|c|c} \mathbf{A}'_{\mathbf{x}'} & \begin{smallmatrix} \mathrm{E} & — & \mathrm{E} \\ | & & | \\ \mathrm{E} & — & \mathrm{E} \end{smallmatrix} & \mathbf{A}_1 \\ \hline \begin{smallmatrix} \mathrm{E} & — & \mathrm{E} \\ | & & | \\ \mathrm{E} & — & \mathrm{E} \end{smallmatrix} & \mathbf{A}''_{\mathbf{x}''} & \mathbf{A}_2 \\ \hline \mathbf{A}'_{\mathbf{x}'} & \mathbf{A}''_{\mathbf{x}''} & \mathbf{A}_3 \end{array}\right] \end{array} \begin{array}{l} \text{candidates (107a) that differ only for features in } \Phi' \\ \text{candidates (107b) that differ only for features in } \Phi'' \\ \text{candidates (107c) that differ for both features in } \Phi'' \text{ and } \Phi'' \end{array}$$

Question (104) can now be made more explicit as follows:

(109) If I know something about the comparative sub-blocks $\mathbf{A}'_{\mathbf{x}'}$ and $\mathbf{A}''_{\mathbf{x}''}$, what can I say about the comparative block $\mathbf{A}_\mathbf{x}$?

Well, not much! But I could say a hell lot more if I got rid of the third vertical block, consisting of $\mathbf{A}_1$, $\mathbf{A}_2$ and $\mathbf{A}_3$. This is the idea pursued in this section.

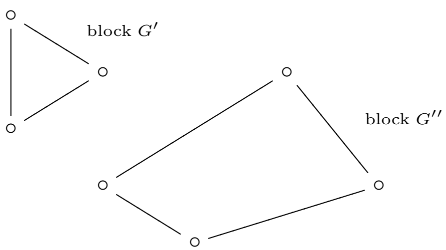
■ **Markedness graph with two components.** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77) corresponding to N features. Assume the constraint set $\mathcal{C}$ is such that:

(110) The feature set can be split into two disjoint sets Φ' and Φ'' s.t. there is no binary markedness constraint in $\mathcal{C}$ that targets both a feature in Φ' and one in Φ''.

Let the *markedness graph* corresponding to a constraint set be as follows:

(111) a. it has N nodes, one for every feature;
b. it has an edge between two nodes iff the constraint set contains a binary markedness constraint that targets the two corresponding features.

Assumption (110) says that the markedness graph can be split into two components G' and G'' with no connections between the two, as in (112).

(112)


In other words again, the two sets of constraints $\mathcal{C}'$ and $\mathcal{C}''$ defined in (103) exhaust the constraint set $\mathcal{C}$ and the comparative block (108) takes the much simplified form in (113).

(113)
$$\mathbf{A}_\mathbf{x} = \begin{array}{c} \begin{array}{cc} \quad\mathcal{C}'\quad & \quad\mathcal{C}''\quad \end{array} \\ \left[\begin{array}{c|c} \mathbf{A}'_{\mathbf{x}'} & \begin{smallmatrix} \mathrm{E} & — & \mathrm{E} \\ | & & | \\ \mathrm{E} & — & \mathrm{E} \end{smallmatrix} \\ \hline \begin{smallmatrix} \mathrm{E} & — & \mathrm{E} \\ | & & | \\ \mathrm{E} & — & \mathrm{E} \end{smallmatrix} & \mathbf{A}''_{\mathbf{x}''} \\ \hline \mathbf{A}'_{\mathbf{x}'} & \mathbf{A}''_{\mathbf{x}''} \end{array}\right] \end{array} \begin{array}{l} \text{candidates (107a) that differ only for features in } \Phi' \\ \text{candidates (107b) that differ only for features in } \Phi'' \\ \text{candidates (107c) that differ for both features in } \Phi'' \text{ and } \Phi'' \end{array}$$

The factorization (113) has two important consequences. Given a ranking $\gg$ on $\mathcal{C}$, let:

(114) a. $\gg'$ = the restriction of $\gg$ to $\mathcal{C}'$;
b. $\gg''$ = the restriction of $\gg$ to $\mathcal{C}''$.

The *first consequence* of the factorization (113) of comparative blocks is that OT-compatibility factorizes over the two factors:

(115)
$$\gg \text{ is OT-compatible with } \mathbf{x} \in \mathcal{X} \iff \begin{cases} \mathbf{x} = (\mathbf{x}', \mathbf{x}'') \\ \gg' \text{ is OT-compatible with } \mathbf{x}' \\ \gg'' \text{ is OT-compatible with } \mathbf{x}'' \end{cases}$$

Given two ranking vectors $\boldsymbol{\theta}_{\mathrm{old}}, \boldsymbol{\theta}_{\mathrm{new}}$ and a comparative row $\mathbf{a}$, split them up:

(116)
$$\begin{aligned} \boldsymbol{\theta}_{\mathrm{old}} &= (\boldsymbol{\theta}'_{\mathrm{old}},\, \boldsymbol{\theta}''_{\mathrm{old}}) \\ \boldsymbol{\theta}_{\mathrm{new}} &= (\boldsymbol{\theta}'_{\mathrm{new}},\, \boldsymbol{\theta}''_{\mathrm{new}}) \end{aligned}$$

The *second consequence* of the factorization (113) of comparative blocks is that the behavior of the OT online algorithm factorizes over the two factors:

(117) For every comparative row $\mathbf{a}$ that belongs to the two top blocks in (113):
$$\boldsymbol{\theta}_{\mathrm{new}} = update_{\mathrm{OT}}(\boldsymbol{\theta}_{\mathrm{old}}, \mathbf{a}) \iff \begin{cases} \boldsymbol{\theta}_{\mathrm{new}} = (\boldsymbol{\theta}'_{\mathrm{new}}, \boldsymbol{\theta}''_{\mathrm{new}}) \\ \boldsymbol{\theta}'_{\mathrm{new}} = update_{\mathrm{OT}}(\boldsymbol{\theta}'_{\mathrm{old}}, \mathbf{a}') \\ \boldsymbol{\theta}''_{\mathrm{new}} = update_{\mathrm{OT}}(\boldsymbol{\theta}''_{\mathrm{old}}, \mathbf{a}'') \end{cases}$$

By (15), I don't need to worry about the case where the current comparative row belongs to the third bottom block, as such a row will never be sampled.

■ **Claim 4** If the OT online algorithm is correct for the factor typologies τ' and τ'' starting from some initial vectors $\boldsymbol{\theta}'_{\mathrm{init}}$ and $\boldsymbol{\theta}''_{\mathrm{init}}$ respectively, then it is correct for the original typology τ starting from the initial vector $\boldsymbol{\theta}_{\mathrm{init}} = (\boldsymbol{\theta}'_{\mathrm{init}}, \boldsymbol{\theta}''_{\mathrm{init}})$.

■ *Proof.* Consider a language L in the typology τ, an L-converging sequence (118a); the corresponding sequence (118b) of ranking vectors entertained by the OT online algorithm starting from $\boldsymbol{\theta}_{\mathrm{init}}$; the corresponding sequence of OT languages (118c).

$$\begin{array}{llcccccc}
\text{(118)} & \text{a.} & & \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_t & \cdots \\
& & & \downarrow & \downarrow & & \downarrow & \\
& \text{b.} & \boldsymbol{\theta}_{\text{init}} \longrightarrow & \boldsymbol{\theta}_1 \longrightarrow & \boldsymbol{\theta}_2 \longrightarrow & \cdots \longrightarrow & \boldsymbol{\theta}_t \longrightarrow & \cdots \\
& & \downarrow & \downarrow & \downarrow & & \downarrow & \\
& \text{c.} & L_{\text{init}} & L_1 & L_2 & \cdots & L_t & \cdots
\end{array}$$

Suppose by contradiction that the OT online algorithm is not correct for τ. This means that for some target language L we have $L_t \not\subseteq L$ in (118), Namely:

(119) There exists some form $\mathbf{x} \in \mathcal{X}$ such that $\mathbf{x} \in L_t$ but $\mathbf{x} \notin L$.

Split up everything:

(120) a. split up the form $\mathbf{x}$ in (118) into $\mathbf{x} = (\mathbf{x}', \mathbf{x}'')$
b. split up each comparative row $\mathbf{a}_t$ in (118a) into $\mathbf{a}_t = (\mathbf{a}'_t, \mathbf{a}''_t)$
c. split up each ranking vector $\boldsymbol{\theta}_t$ in (118b) into $\boldsymbol{\theta}_t = (\boldsymbol{\theta}'_t, \boldsymbol{\theta}''_t)$

Consider now the following rankings:

(121) a. $\gg$ is a ranking that OT-corresponds to target language L, i.e. L;
b. $\gg'$ is the ranking induced on $\mathcal{C}'$ by $\gg$;
c. $\gg''$ is the ranking induced on $\mathcal{C}''$ by $\gg$.

By (121a), $\gg$ is not OT-compatible with the comparative block $\mathbf{A}_{\mathbf{x}}$ corresponding to the form $\mathbf{x}$ in (119). By (115), this means that either $\gg'$ is not OT-compatible with $\mathbf{A}'_{\mathbf{x}'}$ or $\gg''$ is not OT-compatible with the block $\mathbf{A}''_{\mathbf{x}''}$. Wlg, suppose the former case holds:

(122) $\gg'$ is not OT-compatible with $\mathbf{A}'_{\mathbf{x}'}$.

The situation in (118) entails the situation in (123) — I have just added primes in (123).

$$\begin{array}{llcccccc}
\text{(123)} & \text{a.} & & \mathbf{a}'_1 & \mathbf{a}'_2 & \cdots & \mathbf{a}'_t & \cdots \\
& & & \downarrow & \downarrow & & \downarrow & \\
& \text{b.} & \boldsymbol{\theta}'_{\text{init}} \longrightarrow & \boldsymbol{\theta}'_1 \longrightarrow & \boldsymbol{\theta}'_2 \longrightarrow & \cdots \longrightarrow & \boldsymbol{\theta}'_t \longrightarrow & \cdots \\
& & \downarrow & \downarrow & \downarrow & & \downarrow & \\
& \text{c.} & L'_{\text{init}} & L'_1 & L'_2 & \cdots & L'_t & \cdots
\end{array}$$

This can be described as follows: we are training the algorithm on the language L' corresponding to the ranking $\gg'$ within the factor typology τ'; language L' does not contain form $\mathbf{x}'$; yet, $\boldsymbol{\theta}'_t$ is OT-compatible with it; thus (123) contradicts the hypothesis that the OT online algorithm is correct for the factor typology τ'. □

11 Conclusion and applications

■ **Claim 5** Consider a typology $\tau = (\mathcal{X}, Gen, \mathcal{C})$ of the form (66)-(77) corresponding to N features. Assume that:

(124) a. *The mode of feature interaction is "simple":*
namely, no binary markedness constraint punishes a form that is unmarked with respect to both features targeted by that binary markedness constraint.
b. *The amount of feature interaction is "limited":*
namely, there are no two binary markedness constraints that target the same feature.
c. *The set of candidates is rich enough*:
namely, if features i and j interact through a binary markedness constraint, then $\mathcal{X}$ contains forms that realize all possible combinations for x_i and x_j.

Then, the OT online algorithm is correct for the typology τ.

■ *Proof.* By (124b), the typology τ can be split up into a certain number of factor typologies with at most 2 features and no connections between the corresponding components in the markedness graph. Claim 4 thus guarantees that in order for the OT online algorithm to be correct on τ, it is enough that it is correct on each factor typology. By (124a) and (124c), that follows from claim 3. □

■ **Claim 6** Consider the variant of the Azba typology without the distinction between positional and general faithfulness constraints, as follows:

(125) a. $\left\{ \begin{array}{cccc} \text{pa,} & \text{ba,} & \text{sa,} & \text{za,} \\ \text{apsa,} & \text{apza,} & \text{absa,} & \text{abza,} \end{array} \right\}$

b. $\mathcal{C} = \left\{ \begin{array}{ll} F_1 = \text{IDENT[STOP-VOICE]} & M_1 = {}^*\text{[STOP-VOICE]} \\ F_2 = \text{IDENT[FRIC-VOICE]} & M_2 = {}^*\text{[FRIC-VOICE]} \\ & M_3 = \text{AGREE} \end{array} \right\}$

c. $Gen(\text{pa}) = Gen(\text{ba}) = \{\text{pa}, \text{ba}\}$,
$Gen(\text{sa}) = Gen(\text{za}) = \{\text{sa}, \text{za}\}$,
$Gen(\text{apsa}) = Gen(\text{apza}) = Gen(\text{absa}) = Gen(\text{abza}) = \{\text{apsa, apza, absa, abza}\}$

The OT online algorithm is correct on this typology.

■ *Proof.* Follows straightforwardly from claim 5. □

■ **Claim 7** Consider again the vowel typology, repeated below:

(126) a. $\mathcal{X} = \mathcal{Y} = \left\{ \begin{array}{ccc} \text{\#y} & \text{-y-} & \text{y\#} \\ \text{\#i} & \text{-i-} & \text{i\#} \\ \text{\#o} & \text{-o-} & \text{o\#} \\ \text{\#e} & \text{-e-} & \text{e\#} \\ \text{\#A} & \text{-A-} & \text{A\#} \\ \text{\#a} & \text{-a-} & \text{a\#} \end{array} \right\}$

b. $\mathcal{C} = \left\{ \begin{array}{ll} F_1 = \text{IDENT[HIGH]} & M_1 = *[\text{-HIGH, -LOW}] = \{o, e\} \\ F_2 = \text{IDENT[LOW]} & M_2 = *[\text{WORD-FINAL HIGH}] \\ F_3 = \text{IDENT[ROUND]} & M_3 = *[\text{NON-INITIAL, ROUND}] \end{array} \right\}$

c. Gen is an equivalence relation defined by the following classes:

i. #y $\stackrel{\text{Gen}}{\sim}$ #i $\stackrel{\text{Gen}}{\sim}$ #o $\stackrel{\text{Gen}}{\sim}$ #e $\stackrel{\text{Gen}}{\sim}$ #A $\stackrel{\text{Gen}}{\sim}$ #a

ii. -y- $\stackrel{\text{Gen}}{\sim}$ -i- $\stackrel{\text{Gen}}{\sim}$ -o- $\stackrel{\text{Gen}}{\sim}$ -e- $\stackrel{\text{Gen}}{\sim}$ -A- $\stackrel{\text{Gen}}{\sim}$ -a-

iii. #y $\stackrel{\text{Gen}}{\sim}$ i# $\stackrel{\text{Gen}}{\sim}$ o# $\stackrel{\text{Gen}}{\sim}$ e# $\stackrel{\text{Gen}}{\sim}$ A# $\stackrel{\text{Gen}}{\sim}$ a#

The OT online algorithm is correct on this typology.

Proof. Define the set Φ of features as follows:

(127) φ_1 = word-final high
φ_2 = high
φ_3 = low
φ_4 = non-initial round

Assume that:

(128) a. [+HIGH] corresponds to value 1;
b. [-LOW] corresponds to value 1.

The set $\mathcal{X}$ of forms can then be described as follows:

(129)
$$\begin{array}{rlrlrl} \text{\#y} = & [\#, 1, 1, 1] & \text{-y-} = & [\#, 1, 1, 1] & \text{y\#} = & [1, 1, 1, 1] \\ \text{\#i} = & [\#, 1, 1, 0] & \text{-i-} = & [\#, 1, 1, 0] & \text{i\#} = & [1, 1, 1, 0] \\ \text{\#o} = & [\#, 0, 1, 1] & \text{-o-} = & [\#, 0, 1, 1] & \text{o\#} = & [0, 0, 1, 1] \\ \text{\#e} = & [\#, 0, 1, 0] & \text{-e-} = & [\#, 0, 1, 0] & \text{e\#} = & [0, 0, 1, 0] \\ \text{\#A} = & [\#, 0, 0, 1] & \text{-A-} = & [\#, 0, 0, 1] & \text{A\#} = & [0, 0, 0, 1] \\ \text{\#a} = & [\#, 0, 0, 0] & \text{-a-} = & [\#, 0, 0, 0] & \text{a\#} = & [0, 0, 0, 0] \end{array}$$

and the constraint set can be defined as follows, where the binary markedness constraint only punishes forms $\mathbf{x} = (x_1, x_2, x_3, x_4)$ such that $(x_2, x_3) = (1, 1)$.

(130) $\mathcal{C} = \left\{ \begin{array}{ll} F_2 & M_{2,3} \\ F_3 & M_1 \\ F_4 & M_4 \end{array} \right\}$

Split up the set of features as follows:

(131) $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\} = \{\varphi_1\} \cup \{\varphi_2, \varphi_2\} \cup \{\varphi_4\}$

Assumptions (124a) and (124b) are satisfied. Assumption (124c) is not, as there is no form $\mathbf{x} = (x_1, 1, 0, x_4)$. Yet, we can pretend that the set of forms also contain $(\#, 1, 0, \#)$, as this form is unmarked. Correctness of the OT online algorithm thus follows from claim 5. □

References

Berwick, Robert. 1985. *The acquisition of syntactic knowledge*. Cambridge, MA: MIT Press.

Davidson, Lisa, Peter W. Jusczyk, and Paul Smolensky. 2004. "The initial and final states: Theoretical implications and experimental explorations of richness of the base". In *Constraints in Phonological Acquisition*, ed. R. Kager, J. Pater, and W. Zonneveld, 158–203. Cambridge University Press.

Eisner, Jason. 2000. "Easy and Hard Constraint Ranking in Optimality Theory". In *Finite-State Phonology: Proceedings of the Fifth Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, ed. J. Eisner, L. Karttunen, and A. Thériault, 22–33. Luxembourg.

Fikkert, Paula, and Helen De Hoop. 2009. "Language acquisition in optimality theory". *Linguistics* 47.2:311–357.

Galil, Zvi, and Nimrod Megiddo. 1977. "Cyclic Ordering is NP-complete". *Theoretical Computer Science* 5:179–182.

Garey, Michael R., and David S. Johnson. 1979. *Computers and Intractability. A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman and Company.

Hale, Mark, and Charles Reiss. 1998. "Formal and Empirical Arguments Concerning Phonological Acquisition". *Linguistic Inquiry* 29.4:656–683.

Hayes, Bruce. 2004. "Phonological Acquisition in Optimality Theory: The Early Stages". In *Constraints in Phonological Acquisition*, ed. R. Kager, J. Pater, and W. Zonneveld, 158–203. Cambridge University Press.

Heinz, Jeffrey, Gregory M. Kobele, and Jason Riggle. 2009. "Evaluating the Complexity of Optimality Theory". *Linguistic Inquiry* 40:277–288.

Jusczyk, Peter, Paul Smolensky, and Theresa Allocco. 2002. "How English-learning infants respond to Markedness and Faithfulness constraints". *Language Acquisition* 10:31–73.

Manzini, M. Rita, and Ken Wexler. 1987. "Parameters, Binding Theory, and Learnability". *Linguistic Inquiry* 18.3:413–444.

Prince, Alan, and Bruce Tesar. 2004. "Learning Phonotactic Distributions". In *Constraints in Phonological Acquisition*, ed. R. Kager, J. Pater, and W. Zonneveld, 245–291. Cambridge University Press.

Revithiadou, A., and Marina Tzakosta. 2004. "Markedness hierarchies vs. positional faithfulness and the role of multiple grammars in the acquisition of Greek". In *Proceedings of GALA 2003 (Generative Approaches to Language Acquisition)*, ed. S. Baauw and J. Van Kampen, 377–388. Utrecht: LOT Occasional Series. .

Smith, Jennifer. 2000. “Positional faithfulness and learnability in Optimality Theory”. In *Proceedings of ESCOL99*, ed. Rebecca Daly and Anastasia Riehl, 203–214. Ithaca, New York: CLC Publications. .

Smolensky, Paul. 1996a. “On the Comprehension/Production Dilemma in Child Language”. *Linguistic Inquiry* 27.4:720–731.

Smolensky, Paul. 1996b. “The Initial State and Richness of the Base in Optimality Theory”. John Hopkins Technical Report.

Tesar, Bruce. 1995. “Computational Optimality Theory”. Doctoral Dissertation, University of Colorado, Boulder. ROA 90.

Tesar, Bruce. 2008. “Output-Driven Maps”. Ms., Rutgers University; ROA-956.

Tesar, Bruce, and Paul Smolensky. 1998. “Learnability in Optimality Theory”. *Linguistic Inquiry* 29:229–268.

Wareham, Harold Todd. 1998. Systematic Parameterized Complexity Analysis in Computational Phonology. Doctoral Dissertation, University of Victoria, Dept. of Computer Science.