Why phonologists got it right: a principled derivation of OT and HG

Giorgio Magri (SFL, CNRS, University of Paris 8)

1 Introduction

The question concerning the mode of constraint interaction in constraint-based phonology has been mostly neglected so far. Within categorical phonology, the Optimality Theory (OT; Prince & Smolensky 1993/2004) mode of constraint interaction has been adopted without almost any discussion of alternatives. The Harmonic Grammar (HG; Smolensky & Legendre 2006) alternative explored in the most recent literature is only minimally different from OT (as argued in Pater 2009). Within probabilistic phonology, the Maximum Entropy (ME; Hayes & Wilson 2008; Wilson 2025) mode of constraint interaction has been adopted just as unquestioningly. The discussion of alternatives has been limited to Noisy or Stochastic Harmonic Grammar (SHG; Boersma & Pater 2016; Hayes 2017; Magri & Anttila 2025) and based solely on circumstantial evidence about ME's and SHG's relative ability to fit specific patterns of empirical frequencies given specific choices of candidates and constraints (Smith & Pater 2020; Flemming 2021; Breiss & Albright 2022). How can this question of the mode of constraint interaction be tackled in a principled way?

To answer this question, we turn to geometry. It deals with a question analogous to the question of the mode of constraint interaction: among many formally consistent geometric systems, which one best characterizes our physical world? This is an **empirical question** that can only be settled through experimental measurements. But what should be measured? **Formal analyses** of these geometric systems are needed to reduce their differences to a few geometric axioms (say, an axiom on the number of parallel lines) whose validity can then be tested empirically. "The properties which distinguish space from other conceivable triply extended magnitudes are only to be deduced from experience. Thus arises the problem, to discover the simplest matters of fact from which the measure-relations of space may be determined" (Riemann 1873). Although formal analyses cannot settle an empirical question, they can nonetheless be of great help by pinpointing the exact question (say, how many parallel lines?) to be addressed empirically.

The approach I advocate for tackling the problem of the mode of constraint interaction is therefore axiomatic in nature: for each mode of constraint interaction (OT, HG, ME, SOT, SHG), the goal is to prove that it is equivalent to some simple general axiom of phonological behavior. The axiom thus captures explicitly and exhaustively what exactly we commit to when we endorse that mode of constraint interaction. We can determine whether that mode of constraint interaction falls or stands based solely on the empirical validity of the corresponding phonological axioms. In other words, the axioms are "the simplest matters of fact from which the [theory of constraint interaction] may be determined".

This paper sketches this program in the categorical setting for the case OT and HG. Section 2 provides an explicit formulation of the question of the mode of constraint interaction in categorical phonology. Section 3 introduces a very elementary axiom of phonological behavior, left implicit so far in the literature, dubbed the **innocuous concatenation axiom** (ICA). Section 4 shows how categorical constraint-based grammars can flout the ICA without a judicious choice of the mode of constraint interaction. Section 5 finally shows that the OT/HG modes of constraint interaction used in the current phonological literature can be axiomatically derived from the ICA. We can determine whether the OT and HG modes of constraint interaction fall or stand based solely on the empirical validity of the ICA. A companion paper (Magri 2025) extends this axiomatic approach to the mode of constraint interaction from the categorical to the probabilistic setting.

Proceedings of *AMP 2023/2024*

$$Gen = \left\{ \begin{array}{cccc} (/\text{da}/, [\text{ta}]) & (/\text{da}/, [\text{da}]) & & \\ (/\text{adta}/, [\text{atta}]) & (/\text{adta}/, [\text{atda}]) & (/\text{adta}/, [\text{adta}]) & (/\text{adta}/, [\text{adda}]) \\ (/\text{adtada}/, [\text{attata}]) & (/\text{adtada}/, [\text{adtata}]) & (/\text{adtada}/, [\text{atdata}]) & (/\text{adtada}/, [\text{addata}]) \\ (/\text{adtada}/, [\text{attada}]) & (/\text{adtada}/, [\text{adtada}]) & (/\text{adtada}/, [\text{atdada}]) & (/\text{adtada}/, [\text{addada}]) \end{array} \right\} \quad \boldsymbol{C} = \left\{ \begin{array}{c} \text{NoComCoda} \\ \text{AgreeVoice} \\ \text{NoVoice} \\ \text{IdentVoice} \end{array} \right\}$$

Figure 1: The phonological system $Gen, \boldsymbol{C}$ used as the running example of the paper.

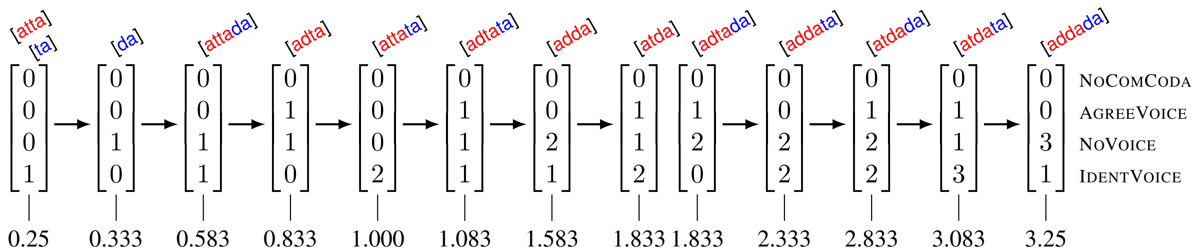

Figure 2: Constraint violation vectors of the mappings in figure 1, ordered according to the SDS order in (1).

2 What is the mode of constraint interaction?

As usual, we start the description of the phonological system of interest by listing into the set Gen all the relevant phonological mappings. We construe a phonological mapping as a pair (x, y) of an underlying form x and a surface realization y. We then construct a set $\boldsymbol{C}$ consisting of a finite number n of constraints $C_1, \ldots, C_n$ that suffice to quantify all the phonologically relevant properties of the mappings listed by Gen. The number of violations assigned by a constraint C_k to a mapping (x, y) is denoted by $C_k(\mathsf{x}, \mathsf{y})$. The constraint violation profile of a mapping (x, y) is the tuple $\boldsymbol{C}(\mathsf{x}, \mathsf{y}) = (C_1(\mathsf{x}, \mathsf{y}), \ldots, C_n(\mathsf{x}, \mathsf{y}))$ of its numbers of constraint violations. Although violation profiles are usually represented as rows of the familiar tableaux, in this paper it will prove more convenient to represent violation profiles as column vectors of numbers.

To illustrate, we consider the phonological system $Gen, \boldsymbol{C}$ in figure 1, that will be used as the running example throughout the paper. Gen lists the underlying strings /adta/ and /da/ together with their concatenation /adtada/. Their candidate surface realizations are obtained by changing underlying obstruent voicing in all possible ways. The constraint set $\boldsymbol{C}$ consists of $n = 4$ constraints: $C_1 =$ NoCompCoda together with three constraints for obstruent voicing, namely $C_2 =$ AgreeVoice (that penalizes surface adjacent obstruents that disagree in voicing), $C_3 =$ NoVoice, and $C_4 =$ IdentVoice. The mappings listed by Gen are then represented as column vectors consisting of the $n = 4$ corresponding numbers of constraint violations, as in figure 2. Each vector is annotated with just the surface candidate, as the corresponding underlying form can be univocally retrieved from the surface candidate.

Numbers can be compared based on their size. Vectors of numbers can therefore be compared based on the size of their components in gazillion different ways. From now on, $\prec$ denotes a (strict, partial, non-trivial) **order** among n-dimensional vectors and $\boldsymbol{x} \prec \boldsymbol{y}$ means that the vector $\boldsymbol{x}$ counts as smaller than the vector $\boldsymbol{y}$ according to this specific way $\prec$ of ordering vectors. To illustrate, for each vector $\boldsymbol{x} = (x_1, \ldots, x_n)$, we square its kth component x_k, obtaining x_k^2; we multiply this square by the coefficient $1/k$ that decreases as k increases, obtaining x_k^2/k; finally, we add these rescaled squares together, obtaining the **sum of decreasing squares** (SDS) $\sum_{k=1}^n x_k^2/k$ of the vector $\boldsymbol{x}$. We can then order two vectors based on the relative size of their SDS, as in (1). To illustrate, the constraint violation vectors of the phonological system $Gen, \boldsymbol{C}$ in figure 1 are then ordered as in figure 2. Throughout this paper, we represent the vector inequality $\boldsymbol{x} \prec \boldsymbol{y}$ by drawing an arrow from $\boldsymbol{x}$ to $\boldsymbol{y}$. For ease of interpretation, each vector in figure 2 is annotated underneath with its SDS.

(1) $\boldsymbol{x} \prec_{\text{SDS}} \boldsymbol{y}$ if and only if $\displaystyle\sum_{k=1}^{n} \frac{x_k^2}{k} < \sum_{k=1}^{n} \frac{y_k^2}{k}$

Let's take stock. We have listed into Gen the pairs of underlying and surface representations that are relevant to describe the phonological system of interest. We have developed a set $\boldsymbol{C}$ of n constraints that quantify the relevant phonological properties of those representations. Finally, we have figured out the relevant way $\prec$ of ordering n-dimensional vectors. It is then natural to define a categorical phonological grammar G as follows. It takes an underlying form x and it returns the set $G(\mathsf{x}) = \{\dots \mathsf{y} \dots\}$ of its surface realizations. We allow for a categorical grammar to return more than a single surface realization to allow for variation. Obviously, a candidate y of the underlying form x makes it to this elite set $G(\mathsf{x})$ only if y is an optimal surface realization of x. Optimal in the sense that Gen lists no other candidate z that would be better than y because z violates the constraints less than y. This assumption that z violates the constraints less than y can be formalized through the inequality $\boldsymbol{C}(\mathsf{x}, \mathsf{z}) \prec \boldsymbol{C}(\mathsf{x}, \mathsf{y})$ that the constraint violation vector of z counts as smaller than that of y according to the relevant way $\prec$ of ordering vectors, as summarized in (2).

(2) $\mathsf{y} \in G(\mathsf{x})$ $\iff$ y is an optimal surface realization of the underlying form x
$\iff$ no other candidate z is a better surface realization of the underlying form x than y
$\iff$ no other candidate z violates the constraints less than y
$\iff$ there exists no candidate z such that $\boldsymbol{C}(\mathsf{x}, \mathsf{z}) \prec \boldsymbol{C}(\mathsf{x}, \mathsf{y})$

The categorical phonological grammar based on the constraint set $\boldsymbol{C}$ through the order $\prec$ as in (2) is denoted as $G^{\boldsymbol{C}}_{\prec}$. To illustrate, the grammar based on the phonological system $Gen, \boldsymbol{C}$ in figure 1 and the SDS-based order $\prec_{\text{SDS}}$ plotted in figure 2 devoices the underlying string /da/ to [ta] and does not allow it to surface faithfully as [da], namely $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}(/\mathsf{da}/) = \{[\mathsf{ta}]\}$. The reason is that the constraint violation vector of [ta] is smaller than that of [da], as shown by the arrow from the former to the latter in figure 2. Indeed, since the squared violations of the faithfulness constraint C_4 = IDENTVOICE are multiplied by the small coefficient $1/4$ while those of the markedness constraint C_3 = NOVOICE are multiplied by the larger coefficient $1/3$, the unfaithful unmarked candidate [ta] violates the constraints less than the faithful marked candidate [da].

Obviously, there are a gazillion different ways $\prec$ of ordering n-dimensional vectors (we will see some more examples in section 4 below). Likely, some yield phonologically crazy constraint-based grammars $G_{\prec}$. Some are instead at least less obviously crazy. The question "what is the mode of constraint interaction?" thus becomes the question "which orders $\prec$ among n-dimensional vectors yield sensible grammars $G_{\prec}$?" In order for the latter question to make sense, we need to make explicit what it means that a categorical grammar is phonologically sensible. The next section makes a concrete proposal.

3 Innocuous concatenation axiom (ICA)

Subsection 3.3 formulates a broad axiom on the behavior of categorical phonological grammars on underling strings that are innocuous in the technical sense of subsection 3.2. Before these technicalities, subsection 3.1 sketches the intuition through three examples.

3.1 *Intuition* As our **first example**, we suppose that some grammar G realizes the underlying string /ad/ faithfully as [ad] and it also realizes the underlying string /tada/ faithfully as [tada]. If we know nothing else about G, can we conclude that G realizes the concatenation /adtada/ of these two underlying strings /ad/ and /tada/ as the concatenation [adtada] of their surface realizations [ad] and [tada] in isolation? Obviously no. In fact, G might ban clusters of obstruents that disagree in voicing (penalized by the constraint AGREEVOICE mentioned above) and one such marked structure has been **created** by concatenating ad and tada into adtada.

As our **second example**, we suppose that some grammar G neutralizes the underlying string /adt/ to [ad] by simplifying its complex coda and it realizes the underlying string /ada/ faithfully as [ada]. If we know nothing else about G, can we conclude that G realizes the concatenation /adtada/ of these two underlying strings /adt/ and /ada/ as the concatenation [adada] of their surface realizations [ad] and [ada] in isolation? Obviously no. In fact, G might ban complex codas (penalized by the constraint NOCOMCODA mentioned above) and one such marked structure is **dissolved** (through proper syllabification that is left implicit here for ease of exposition) by concatenating adt and ada into adtada.

As our **third example**, we suppose that some grammar G realizes the underlying string /adta/ faithfully as [adta] and it also realizes the underlying string /da/ faithfully as [da]. If we know nothing else about G, can we conclude that G realizes the concatenation /adtada/ of these two underlying strings /adta/ and /da/ as the concatenation [adtada] of their surface realizations [adta] and [da] in isolation? Probably yes. In fact, the concatenation of adta and da into adtada plausibly does not create nor dissolve any relevant marked structures (at least, no marked structures penalized by the relevant markedness constraints listed in figure 1)

3.2 *Technicalities* We now want to formalize the difference between the first two examples versus the third one. To this end, we start by saying that a grammar G is **concatenative** on a pair of underlying strings x' and x'' provided the surface realizations of the concatenation $\mathsf{x}' \cdot \mathsf{x}''$ of those underlying strings is the concatenation of the surface realizations of x' and x'' in isolation, as stated in (3).[1] In other words, grammar application respects string concatenation because the two operations commute: we get the same result when we first concatenate and then apply the grammar (as prescribed by the left-hand side) versus when we first apply the grammar and then concatenate (as prescribed by the right-hand side).

(3)
$$G(\underbrace{\mathsf{x}' \cdot \mathsf{x}''}_{\text{concatenation of underlying strings}}) = \underbrace{G(\mathsf{x}') \cdot G(\mathsf{x}'')}_{\text{concatenation of surface strings}}$$

To illustrate, a grammar G is concatenative on the pair of underlying strings /adta/ and /da/ provided it satisfies the identity $G(/\mathsf{adtada}/) = G(/\mathsf{adta}/) \cdot G(/\mathsf{da}/)$. For concreteness, if G realizes both underlying strings /adta/ and /da/ in isolation faithfully as [adta] and [da], then G also realizes the concatenation /adtada/ of the two underlying strings faithfully as the concatenation [adtada] of their faithful surface realizations in isolation. Section 4 below discusses more examples of compliance with and failure of concatenativity.

When should a grammar satisfy this concatenativity condition (3)? To address this question, we say that the concatenation $\mathsf{y}' \cdot \mathsf{y}''$ of two surface strings y' and y'' is **innocuous** relative to the relevant set of markedness constraints provided the following two conditions hold. The first condition is that the concatenation $\mathsf{y}' \cdot \mathsf{y}''$ does not **create** any relevant markedness violations in the sense that the number $M(\mathsf{y}' \cdot \mathsf{y}'')$ of violations assigned by any relevant markedness constraint M to the concatenation is not larger than the sum of the numbers $M(\mathsf{y}')$ and $M(\mathsf{y}'')$ of markedness violations of the two surface strings in isolation, as stated in (4a). The second condition is that the concatenation $\mathsf{y}' \cdot \mathsf{y}''$ does not **dissolve** any relevant markedness violations in the sense that the number $M(\mathsf{y}' \cdot \mathsf{y}'')$ of violations assigned by any relevant markedness constraint M to the concatenation is not smaller than the sum of the numbers $M(\mathsf{y}')$ and $M(\mathsf{y}'')$ of markedness violations of the two surface strings in isolation, as stated in (4b). It follows that the number of markedness violations of the surface concatenation $\mathsf{y}' \cdot \mathsf{y}''$ is exactly equal to the sum of the numbers of markedness violations of the two strings y' and y'' in isolation for any relevant markedness constraint M, as stated in (4c).

(4) (a) $M(\mathsf{y}' \cdot \mathsf{y}'') \not> M(\mathsf{y}') + M(\mathsf{y}'')$
(b) $M(\mathsf{y}' \cdot \mathsf{y}'') \not< M(\mathsf{y}') + M(\mathsf{y}'')$
(c) $M(\mathsf{y}' \cdot \mathsf{y}'') = M(\mathsf{y}') + M(\mathsf{y}'')$

Finally, we say that the concatenation $\mathsf{x}' \cdot \mathsf{x}''$ of two underlying strings x' and x'' is **innocuous** provided this condition (4c) holds for any candidate surface realizations y' and y'' of the underlying strings x' and x'' in isolation. To illustrate, let us suppose that the relevant markedness constraints are NoComCoda, AgreeVoice, and NoVoice, mentioned above. Figure 3 shows that for any candidate y' of the underlying string /adta/ and for any candidate y'' of the underlying string /da/, the number of markedness violations of the concatenation $\mathsf{y}' \cdot \mathsf{y}''$ is exactly equal to the sum of the numbers of markedness violations of the two candidates y' and y'' in isolation. We conclude that the concatenation of these underlying strings /adta/ and /da/ is innocuous relative to the relevant markedness constraints considered here.

3.3 *Formulating the axiom* The intuition sketched in subsection 3.1 can now be formalized into the axiom boxed below. I submit that this ICA makes explicit an assumption that has always been adopted implicitly in the daily business of phonological analysis. Indeed, it is the ICA that makes it possible to compile a descriptive phonological grammar of a language by only describing the behavior of the grammar on the circumscribed set of non-innocuous concatenations, while tacitly ignoring the large majority of innocuous concatenations. Furthermore, SPE grammars comply with the ICA by design, because a rule $A \rightarrow B / X\_Y$ only applies when the corresponding markedness constraint $*XAB$ is violated. The constraint-based grammars reviewed in section 2 can instead flout the ICA, unless we choose the order $\prec$ used to optimize constraint violations carefully, as shown in section 4 below. This observation will motivate the problem of characterizing the orders that comply with the ICA, that will be tackled in section 5.

[1] The identity in (3) holds between sets: it says that $G(\mathsf{x}' \cdot \mathsf{x}'')$ is the set of all and only the surface strings obtained by concatenating a string from the set $G(\mathsf{x}')$ with a string from the set $G(\mathsf{x}'')$.

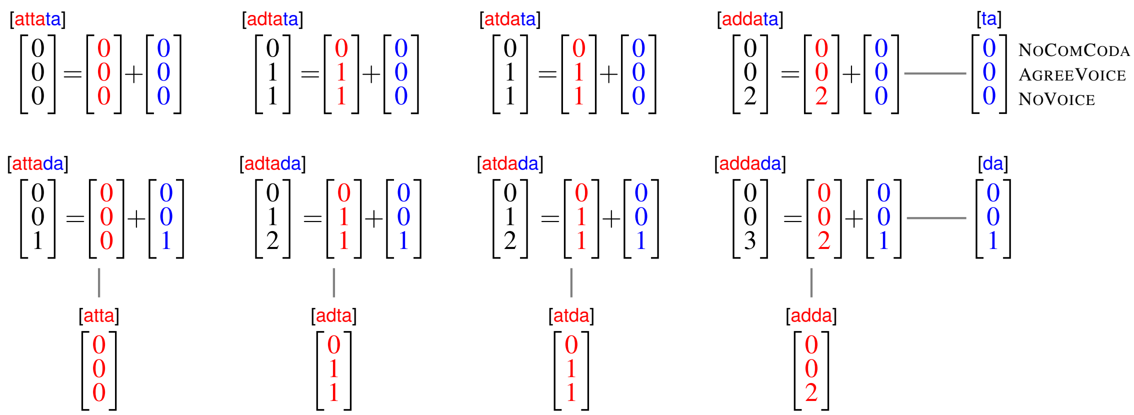

Figure 3: The concatenation of the underlying strings /adta/ and /da/ is innocuous relative to the phonological system $Gen, \boldsymbol{C}$ in figure 1.

> ***Innocuous Concatenation Axiom*** *(ICA): a phonological grammar G must be concatenative on any pair of underlying strings x′ and x′′ whose concatenation is innocuous relative to the markedness constraints that are relevant for the description of the phonological system of interest.*

We now revisit the three examples discussed informally in subsection 3.1. The concatenation of the underlying strings /ad/ and /tada/ considered in the **first example** is not innocuous. In fact, the markedness constraint M = AGREEVOICE is relevant. Yet, the concatenation of the two candidate faithful surface realizations [ad] and [tada] creates a violation of this constraint, thus flouting the innocuousness condition (4a), as stated in (5). The ICA thus allows a categorical grammar G to flout concatenativity on these two underlying strings, namely $G(/\text{adtada}/) \neq G(/\text{ad}/) \cdot G(/\text{tada}/)$.

(5) $\underbrace{M([\text{adtada}])}_{1} > \underbrace{M([\text{ad}])}_{0} + \underbrace{M([\text{tada}])}_{0}$

Also the concatenation of the underlying strings /adt/ and /ada/ considered in the **second example** is not innocuous. In fact, the markedness constraint M = NOCOMCODA is relevant. Yet, the concatenation of the two candidate faithful surface realizations [adt] and [ada] dissolves a violation of this constraint, thus flouting the innocuousness condition (4b), as shown in (6). The ICA thus allows a categorical grammar G to flout concatenativity on these two underlying strings, namely $G(/\text{adtada}/) \neq G(/\text{adt}/) \cdot G(/\text{ada}/)$.

(6) $\underbrace{M([\text{adtada}])}_{0} < \underbrace{M([\text{adt}])}_{1} + \underbrace{M([\text{ada}])}_{0}$;

The concatenation of the underlying strings /adta/ and /da/ considered in the **third example** is instead innocuous, as verified in figure 3. The ICA thus does require a categorical grammar G to satisfy the concatenativity identity $G(/\text{adtada}/) = G(/\text{adta}/) \cdot G(/\text{da}/)$ for these two underlying strings.

3.4 *A note on the faithfulness constraints* Innocuousness has been defined in subsection 3.2 only in terms of the markedness constraints, while the faithfulness constraints have played no role. The reason is that (most) faithfulness constraints satisfy an identity analogous to the markedness identity (4c) and they do so for any concatenations, irrespectively of innocuousness. A full discussion of this issue is postponed to a forthcoming longer version of this paper. Here is a brief sketch of the idea.

So far, we have construed a phonological mapping simply as a pair (x, y) of an underlying string x and a corresponding surface realization y. Following McCarthy & Prince (1995), a phonological mapping should be construed more precisely as a triplet $(\mathsf{x}, \mathsf{y}, R)$ that also includes a **correspondence relation** R between underlying and surface segments. We can then say that a faithfulness constraint F is **additive** provided it

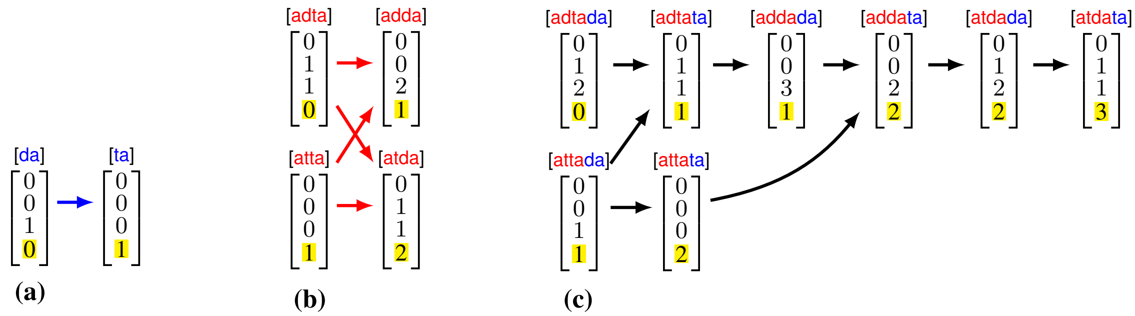

Figure 4: Ordering of the constraint violation vectors of the mappings in figure 1 according to the order $\prec_S$ when the set S of critical constraints consists of only C_4 = IDENTVOICE (whose entries are highlighted).

satisfies the identity (7) for any two mappings $(\mathsf{x}', \mathsf{y}', R')$ and $(\mathsf{x}'', \mathsf{y}'', R'')$. The righthand side features the sum of the numbers of faithfulness violations of these two mappings. The lefthand side features the number of faithfulness violations of the mapping $(\mathsf{x}' \cdot \mathsf{x}'', \mathsf{y}' \cdot \mathsf{y}'', R' \cup R'')$ whose underlying string is the concatenation of the two underlying strings x' and x'', whose surface string is the concatenation of the two surface strings y'' and y'', and whose correspondence relation is the union of the two correspondence relations R' and R''.

(7) $F(\mathsf{x}' \cdot \mathsf{x}'', \mathsf{y}' \cdot \mathsf{y}'', R' \cup R'') = F(\mathsf{x}', \mathsf{y}', R') + F(\mathsf{x}'', \mathsf{y}'', R'')$

This faithfulness identity (7) is formally analogous to the markedness identity (4c) used above to define innocuousness. Furthermore, most "basic" faithfulness constraints (such as MAX and DEP; IDENT$_\varphi$ for any feature φ; featural MAX$_{[+\varphi]}$ and DEP$_{[+\varphi]}$; UNIFORMITY and INTEGRITY; LINEARITY, MAXLINEARITY, and DEPLINEARITY) satisfy this identity (7) no matter the choice of the two mappings on the righthand side.

4 The ICA in constraint-based phonology

This section tests the constraint-based grammars $G_\prec$ formalized in section 2 on the ICA motivated in section 3. These examples show that these constraint-based grammars can flout the ICA. Furthermore, they show that these grammars that flout the ICA are not "unreasonable", although they seem not to reason in the way natural language phonology does.

4.1 *First example* We single out into the set $S \subseteq \{1, \ldots, n\}$ the (indices of the) constraints that we deem **critical** to describe the phonological system of interest. We then define the order $\prec_S$ among n dimensional vectors as in (8). Condition (8a) requires every critical component of the smaller vector $\boldsymbol{x}$ to be indeed smaller than the corresponding component of the larger vector $\boldsymbol{y}$. Non-critical components of the smaller vector $\boldsymbol{x}$ can instead be larger than the corresponding components of the larger vector $\boldsymbol{y}$. Yet, they cannot be too much larger, because condition (8b) requires the sum of all the components of the smaller vector $\boldsymbol{x}$ to be smaller than the sum of all the components of the larger vector $\boldsymbol{y}$. Finally, condition (8c) requires one of the inequalities in (8a) or in (8b) to hold strict, ensuring that the order $\prec_S$ is indeed strict (no vector can be smaller than itself).

(8) $\boldsymbol{x} \prec_S \boldsymbol{y}$ if and only if
(a) $x_k \leq y_k$ for every $k \in S$;
(b) $x_1 + \cdots + x_n \leq y_1 + \cdots + y_n$;
(c) one of the inequalities in (a) or (b) holds strict.

To illustrate, we consider again the phonological system $Gen, \boldsymbol{C}$ in figure 1. We suppose that the set S of critical constraints consists of only the faithfulness constraint, namely $S = \{C_4 = \text{IDENTVOICE}\}$. We want to check whether the grammar $G^{\boldsymbol{C}}_{\prec_S}$ corresponding to the resulting order $\prec_S$ complies with the ICA. To this end, we consider the two underlying strings /adta/ and /da/, whose concatenation has been verified in figure 3 to be innocuous relative to the markedness constraints listed by the constraint set $\boldsymbol{C}$. And we check whether the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying concatenation /adtada/ as the concatenation of the surface realizations of the two underlying strings /adta/ and /da/ in isolation, as required by the ICA.

To start, we compute the surface realizations of the underlying string /da/ in isolation. The constraint violation vectors of its two candidates [da] and [ta] are ordered by $\prec_S$ as in figure 4a. The constraint violation vector of [da] is smaller than that of [ta] (as indicated by the arrow from the former to the latter): the candidate [da] satisfies the critical constraint C_4 = IDENTVOICE and thus violates the constraints less than the candidate [ta] that instead violates the critical constraint C_4 = IDENTVOICE to satisfy the non-critical constraint C_3 = NOVOICE. We conclude that the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying string /da/ faithfully, namely $G_{\prec_S}(/\text{da}/) = \{[\text{da}]\}$.

Next, we compute the surface realizations of the underlying string /adta/ in isolation. The constraint violation vectors of its four candidates are ordered by $\prec_S$ as in figure 4b. The constraint violation vector of [atta] is not smaller than the vector of [adta] because of clause (8a): the faithful candidate [adta] does not violate the constraints too much because the faithfulness violations are deemed critical. Furthermore, the constraint violation vector of [adta] is not smaller than the vector of [atta] because of clause (8b): the mildly unfaithful candidate [atta] does not violate the constraints too much because it achieves a staggering improvement in markedness. Finally, the constraint violation vectors of [adta] and [atta] are each smaller than each of the vectors of the other two candidates [atda] and [adda]: unfaithful candidates [atda] and [adda] that achieve no improvement in markedness violate the constraints too much. We conclude that the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying string /adta/ as either [adta] or [atta], namely $G^{\boldsymbol{C}}_{\prec_S}(/\text{adta}/) = \{[\text{adta}], [\text{atta}]\}$.

Finally, we compute the surface realizations of the underlying string /adtada/. The constraint violation vectors of its eight candidates are ordered by $\prec_S$ as in figure 4c. This ordering can be made sense of by reasoning as above. In particular, despite the faithful constraint being the only critical constraint, the constraint violation vector of the faithful candidate [adtada] does not count as smaller than the vector of the mildly unfaithful candidate [attada] because of clause (8b). We conclude, that the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying string /adtada/ as either [adtada] or [attada], namely $G_{\prec_S}(/\text{adtada}/) = \{[\text{adtada}], [\text{attada}]\}$.

Since the concatenation of the underlying strings /adta/ and /da/ is innocuous, the ICA requires that the set $G^{\boldsymbol{C}}_{\prec_S}(/\text{adtada}/)$ of surface realizations of their concatenation /adtada/ consist of all and only the concatenations of the surface realizations of the two underlying strings in isolation collected into the two sets $G^{\boldsymbol{C}}_{\prec_S}(/\text{adta}/)$ and $G^{\boldsymbol{C}}_{\prec_S}(/\text{da}/)$. This requirement is satisfied, as shown in (9). We conclude that this grammar $G^{\boldsymbol{C}}_{\prec_S}$ complies with the ICA on the two underlying strings considered.

(9) $$\underset{\{[\text{adtada}],[\text{attada}]\}}{G^{\boldsymbol{C}}_{\prec_S}(/\text{adtada}/)} = \underset{\{[\text{adta}],[\text{atta}]\}}{G^{\boldsymbol{C}}_{\prec_S}(/\text{adta}/)} \cdot \underset{\{[\text{da}]\}}{G^{\boldsymbol{C}}_{\prec_S}(/\text{da}/)}$$

4.2 *Second example* We consider again the order $\prec_S$ in (8) but we now enlarge the critical set S to contain not only the faithful constraint C_4 = IDENTVOICE as above but also the markedness constraint C_3 = NOVOICE, namely $S = \{C_3 = \text{NOVOICE}, C_4 = \text{IDENTVOICE}\}$. Again, we want to check whether the grammar $G^{\boldsymbol{C}}_{\prec_S}$ corresponding to the resulting order $\prec_S$ complies with the ICA on the innocuous concatenation of the two underlying strings /adta/ and /da/.

To start, we observe that the constraint violation vectors of the two candidates [da] and [ta] of the underlying string /da/ are mutually unordered, as shown in figure 5a (that indeed has no arrows). Since the set S of critical constraints consists of both the markedness constraint NOVOICE (violated by [da] but not by [ta]) and the faithfulness constraint IDENTVOICE (violated by [ta] but not by [da]), neither of the two candidates violates the constraints more than the other. We conclude that the grammar $G^{\boldsymbol{C}}_{\prec}$ realizes the underlying string /da/ as either [ta] or [da], namely $G_{\prec_S}(/\text{da}/) = \{[\text{da}], [\text{ta}]\}$.

Next, we observe that the constraint violation vectors of the four candidates of the underlying string /adta/ are ordered in figure 5b in exactly the same way as they are ordered in figure 4b, as we can reason in the same way in the two cases. We conclude that also in this second example, the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying string /adta/ as either the faithful candidate [adta] or the unmarked and mildly unfaithful candidate [atta], namely $G_{\prec_S}(/\text{adta}/) = \{[\text{adta}], [\text{atta}]\}$.

Finally, we observe that the constraint violation vectors of the eight candidates of the underlying concatenation /adtada/ are ordered as in figure 5c. The vectors of the three surface concatenations [adtada], [attada], and [attata] listed in the leftmost column of the figure are not smaller than each other, because of clause (8a): each of these three vectors has one of the two critical components which is too large. Furthermore, the constraint violation vector of [attada] (with the first stop devoiced) is smaller than the constraint violation vector of [adtata] (with the third stop devoiced). Formally, the reason is that the two vectors share the same critical components but the latter vector has an additional non-zero entry corresponding to the non-critical constraint C_2 = AGREEVOICE, yielding a larger sum of all components. Phonologically,

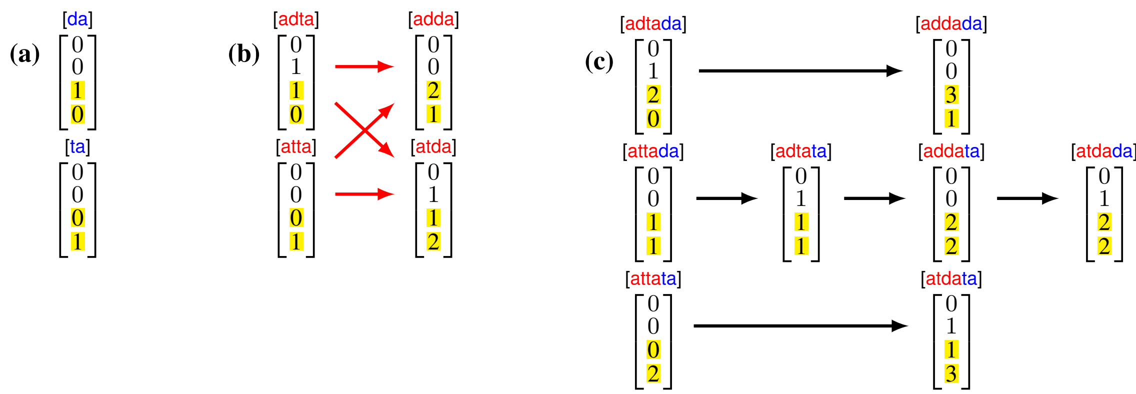

Figure 5: Ordering of the violation vectors of the mappings in figure 1 according to $\prec_S$ when the set S of critical constraints consists of C_3 = NOVOICE and C_4 = IDENTVOICE (whose entries are highlighted).

[attada] is better than [adtata] because, if you can shoot at only one voiced stop (because you can violate the critical faithfulness constraint only once), you should aim carefully: you should devoice the first stop as in [attada], thus repairing a violation of AGREEVOICE as well; you should not devoice the third stop as in [adtata], which yields no improvement with respect to AGREEVOICE. We conclude that the grammar $G^{\boldsymbol{C}}_{\prec_S}$ realizes the underlying concatenation /adtada/ faithfully as [adtada], or by devoicing all stops as [attata], or by devoicing only the first stop as [attada], namely $G_{\prec_S}$(/adtada/) = {[adtada], [attata], [attada]}.

Since the concatenation of the underlying strings /adta/ and /da/ is innocuous, the ICA requires that the set $G^{\boldsymbol{C}}_{\prec_S}$(/adtada/) of surface realizations of their concatenation /adtada/ consist of all and only the concatenations of the surface realizations of the two underlying strings in isolation collected in the two sets $G^{\boldsymbol{C}}_{\prec_S}$(/adta/) and $G^{\boldsymbol{C}}_{\prec_S}$(/da/). This requirement is not satisfied, as shown in (10): the righthand side includes the concatenation [adtata] of the surface strings [adta] with [ta]; the lefthand side instead misses this surface realization [adtata] that does not not aim carefully and thus devoices the "wrong" stop.

(10) $G_{\prec_S}$(/adtada/) $\neq$ $G_{\prec_S}$(/adta/) $\cdot$ $G_{\prec_S}$(/da/)

{ [adtada] [attada] [attata] } {[adta], [atta]} { [da] [ta] }

We conclude that this grammar $G^{\boldsymbol{C}}_{\prec_S}$ flouts the ICA on the two underlying strings /adta/ and /da/ considered. This example shows that the ICA does not follow from the sheer constraint-based architecture, unless we restrict this architecture with suitable assumptions on the orders used to compare constraint violation vectors. Furthermore, this example shows that grammars that flout the ICA are not "unreasonable": there is nothing unreasonable in the choice made by this grammar $G^{\boldsymbol{C}}_{\prec_S}$ to shoot carefully at the right stop and thus to avoid the problematic surface realization [adtata] that devoices the wrong stop. Yet, this type of reasoning does not seem germane to natural language phonology.

4.3 *Third example* We go back to the SDS order $\prec_{\text{SDS}}$ in (1), already plotted in figure 2. Again, we want to check whether the grammar $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}$ corresponding to this order $\prec_{\text{SDS}}$ complies with the ICA on the innocuous concatenation of the two underlying strings /adta/ and /da/. To start, we recall from section 2 that the constraint violation vectors of the two candidates of the underlying string /da/ are ordered as in figure 6a, whereby the grammar $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}$ devoices the underlying string /da/ to [ta], namely $G_{\prec_{\text{SDS}}}$(/da/) = {[ta]}.

Next, we observe that the constraint violation vectors of the four candidates of the underlying string /adta/ are ordered as in figure 6b. The candidate [atta] violates the constraints the least because it only violates the faithfulness constraint C_4 = IDENTVOICE that is multiplied by the smallest coefficient 1/4 and therefore counts the least. The grammar $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}$ thus also devoices the underlying string /adta/ to [atta], namely $G_{\prec_{\text{SDS}}}$(/adta/) = {[atta]}.

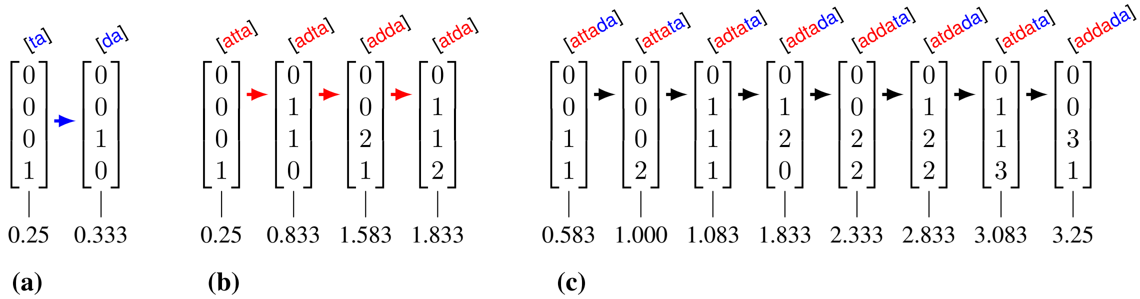

Figure 6: Ordering of the violation vectors of the mappings in figure 1 according to the SDS-order in (1).

Finally, we observe that the constraint violation vectors of the eight candidates of the underlying concatenation /adtada/ are ordered as in figure 6c. The constraint violation vector of the partially devoiced and thus partially marked candidate [attada] is smaller than the vector of the fully devoiced and thus completely unmarked candidate /attata/. The reason is that constraint violations are squared in the computation of SDSs. Thus, one violation of C_3 = NoVoice only contributes $1 \cdot \frac{1}{3}$ to the SDS and one violation of C_4 = IdentVoice only contributes $1 \cdot \frac{1}{4}$, whereby the SDS of [attada] is small, namely equal to $\frac{1}{3} + \frac{1}{4} = 0.583$. Because of squaring, two violations of C_4 = IdentVoice instead contributes $\frac{1}{4} \cdot 2^2 = 1$ to the SDS, whereby the SDS of [attata] is large. The grammar $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}$ thus only partially devoices the underlying concatenation /adtada/ to [attada], namely $G_{\prec_{\text{SDS}}}$(/adtada/) = {[attada]}.

We conclude that the grammar $G^{\boldsymbol{C}}_{\prec_{\text{SDS}}}$ does not satisfy the ICA on the innocuous concatenation of the underlying strings /adta/ and /da/, as stated in (11). Indeed, since constraint violations are squared, the fate of the underlying string /da/ cannot be predicted independently of what it is concatenated with. When /da/ is realized in isolation, it is devoiced to [ta], because a single faithfulness violation is tolerable. The situation is different when instead /da/ is concatenated with /adta/, despite the concatenation being innocuous. In fact, since /adta/ is also devoiced to [atta], the grammar prefers to realize /da/ faithfully in this case, as it would otherwise incur 2 faithfulness violations that, when squared, contribute a total of 4 to the SDS, making it intolerably large.

(11) $G_{\prec_S}$(/adtada/) $\neq$ $G_{\prec_S}$(/adta/) $\cdot$ $G_{\prec_S}$(/da/)

{ [attada] } {[atta]} { [ta] }

5 A characterization of the constraint-based grammars that satisfy the ICA

Let's take stock. Section 1 has argued that constraint-based phonology has so far mostly ignored the question of the mode of constraint interaction. Two modes of constraint interaction have been adopted unquestioningly, without much debate of possible alternatives: the OT mode of constraint interaction has dominated most of the literature; the HG alternative has been explored only in the most recent literature. Section 2 has then formulated the question of the mode of constraint interaction explicitly as the question of which orders $\prec$ among n-dimensional vectors yield grammars $G^{\boldsymbol{C}}_{\prec}$ that are sensible, irrespectively of the choice of the constraint set $\boldsymbol{C}$. Section 3 has then argued that a sensible grammar must in particular comply with the ICA: it must realize the innocuous concatenation of two underlying strings as the concatenation of their surface realizations in isolation. Section 4 has shown that, while the ICA was essentially hardwired into SPE, constraint-based grammars $G^{\boldsymbol{C}}_{\prec}$ can flout the ICA without a judicious choice of the orders $\prec$ used to compare constraint violation vectors. We now have our work cut out for us: we need to characterize those orders $\prec$ among n-dimensional vectors that yield grammars $G^{\boldsymbol{C}}_{\prec}$ that comply with the ICA, irrespectively of the choice of the constraint set $\boldsymbol{C}$. The following box offers this characterization. The proof of this result is deferred to a forthcoming longer version of this paper.

An order $\prec$ among n-dimensional vectors yields a constraint-based grammar $G^{\boldsymbol{C}}_{\prec}$ that complies with the ICA no matter the choice of the constraint set $\boldsymbol{C}$ (whose faithfulness constraints are all additive in the sense of subsection 3.4) if and only if there exist a certain number d (between 1 and n) of n-dimensional **weight vectors**

(12)
$$\begin{aligned} \boldsymbol{w}^{(1)} &= (w^{(1)}_1, \ldots, w^{(1)}_n) \\ &\vdots \\ \boldsymbol{w}^{(d)} &= (w^{(d)}_1, \ldots, w^{(d)}_n) \end{aligned}$$

such that any two vectors $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, \ldots, y_n)$ satisfy the inequality $\boldsymbol{x} \prec \boldsymbol{y}$ if and only if there exists some index i (between 1 and d, that depends on $\boldsymbol{x}$ and $\boldsymbol{y}$) such that:

(13) (a) when we use the first $i-1$ weight vectors $\boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(i-1)}$, the weighted sum of the components of $\boldsymbol{x}$ is equal to the weighted sum of the components of $\boldsymbol{y}$:

$$\begin{aligned} \sum_{k=1}^{n} w^{(1)}_k x_k &= \sum_{k=1}^{n} w^{(1)}_k y_k \\ &\vdots \\ \sum_{k=1}^{n} w^{(i-1)}_k x_k &= \sum_{k=1}^{n} w^{(i-1)}_k y_k \end{aligned}$$

(b) when we use instead the ith weight vector $\boldsymbol{w}^{(i)}$, the weighted sum of the components of $\boldsymbol{x}$ is strictly smaller than the weighted sum of the components of $\boldsymbol{y}$:

$$\sum_{k=1}^{n} w^{(i)}_k x_k < \sum_{k=1}^{n} w^{(i)}_k y_k$$

The complexity of the architecture in (13) motivated by the ICA is controlled by two parameters. One parameter is **depth**, namely the number d of weight vectors $\boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(d)}$ listed in (12). Intuitively, the complexity of the architecture increases as the number d of vectors used increases. The other parameter is **sparsity**, namely the largest number s of non-zero components per weight vector. Intuitively, the complexity of the architecture increases as the number s of weights that are "active" because different from zero increases. A **simple** but **non-trivial** implementation of this architecture can thus be obtained by choosing the minimum value for one of these two parameters (to achieve simplicity), while allowing the other parameter to take the maximum value (to achieve non-triviality). We thus have two simplest non-trivial implementations of the architecture in (13). Let's take a closer look at them in turn.

To start, we enforce minimum depth $d = 1$, whereby we avail ourselves of a unique weight vector $\boldsymbol{w}^{(1=d)}$, that we denote more conveniently as $\boldsymbol{w}$. Yet, we allow for maximum sparsity $s = n$, whereby this unique weight vector $\boldsymbol{w}$ can have all non-zero components. The constraint-based grammar $G^{\boldsymbol{C}}_{\prec}$ corresponding to the order $\prec$ in (13) can then be made explicit as in (14) for any underlying form x and any surface realization y from the candidate set $Gen(\mathsf{x})$. Step (14i) holds because of the general definition (2) of constraint-based grammars. Step (14ii) holds because condition (13a) is trivial when $d = 1$, whereby the definition (13) of the order $\prec$ reduces to just condition (13b). We conclude that this constraint-based grammar $G^{\boldsymbol{C}}_{\prec}$ is the familiar **HG grammar** corresponding to the weight vector $\boldsymbol{w}$.

(14)
$$\begin{aligned} \mathsf{y} \in G^{\boldsymbol{C}}_{\prec}(\mathsf{x}) &\overset{(i)}{\Longleftrightarrow} \text{there exists no other candidate } \mathsf{z} \text{ such that } \boldsymbol{C}(\mathsf{x}, \mathsf{z}) \prec \boldsymbol{C}(\mathsf{x}, \mathsf{y}) \\ &\overset{(ii)}{\Longleftrightarrow} \text{there exists no other candidate } \mathsf{z} \text{ such that } \sum_{k=1}^{n} w_k C_k(\mathsf{x}, \mathsf{z}) < \sum_{k=1}^{n} w_k C_k(\mathsf{x}, \mathsf{y}) \end{aligned}$$

Next, we allow maximum depth $d = n$, whereby we avail ourselves of a full stack (12) of n weight vectors $\boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(d=n)}$. Yet, we enforce minimum sparsity $s = 1$, whereby each of these n vectors has a unique component different from zero. We can assume without loss of generality that the unique non-zero component per weight vector is equal to one. Furthermore, we can assume without loss of generality that no two weight vectors $\boldsymbol{w}^{(i)}$ and $\boldsymbol{w}^{(j)}$ share the same non-zero component.

As a result, this list $\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, \ldots, \boldsymbol{w}^{n}$ of weight vectors in (12) effectively defines the constraint ranking $\gg$ in (15) as follows. The top slot in the ranking is occupied by the constraint C_{k_1} such that k_1 is the index of the unique non-zero component of the first weight vector $\boldsymbol{w}^{(1)}$ in the list. The second slot in the ranking is occupied by the constraint C_{k_2} such that k_2 is the index of the unique non-zero component of the second weight vector $\boldsymbol{w}^{(2)}$ in the list. And so on until the bottom slot in the ranking, which is occupied by the constraint C_{k_n} such that k_n is the index of the unique non-zero component of the last weight vector $\boldsymbol{w}^{(n)}$.

(15) $C_{k_1} \gg C_{k_2} \gg \cdots \gg C_{k_n}$

The constraint-based grammar $G^{\boldsymbol{C}}_{\prec}$ corresponding to the order $\prec$ in (13) can then be made explicit as in (16) for any underlying form x and any surface realization y from the candidate set $Gen(\mathsf{x})$. Step (16i) holds because of the general definition (2) of constraint-based grammars. Step (16ii) holds because the order $\prec$ is defined through condition (13), which is repeated here with the two arbitrary vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ replaced by the two constraint violation vectors $\boldsymbol{C}(\mathsf{x},\mathsf{z})$ and $\boldsymbol{C}(\mathsf{x},\mathsf{y})$. Step (16iii) holds because of the assumption that the unique non-zero component of the weight vector $\boldsymbol{w}^{(i)}$ is its k_ith component, which is furthermore equal to one. Finally, Step (16iv) holds because of the definition of the ranking $\gg$ in (15). We conclude that this constraint-based grammar $G_{\prec}$ is the familiar **OT grammar** corresponding to this constraint ranking $\gg$.

(16) $\mathsf{y} \in G^{\boldsymbol{C}}_{\prec}(\mathsf{x}) \iff$

$\overset{(i)}{\iff}$ there exists no other candidate z such that $\boldsymbol{C}(\mathsf{x},\mathsf{z}) \prec \boldsymbol{C}(\mathsf{x},\mathsf{y})$

$\overset{(ii)}{\iff}$ there exist no other candidate z and no index i such that

$$(a)\quad \sum_{k=1}^{n} w_k^{(1)} C_k(\mathsf{x},\mathsf{z}) = \sum_{k=1}^{n} w_k^{(1)} C_k(\mathsf{x},\mathsf{y}) \quad \vdots \quad \sum_{k=1}^{n} w_k^{(i-1)} C_k(\mathsf{x},\mathsf{z}) = \sum_{k=1}^{n} w_k^{(i-1)} C_k(\mathsf{x},\mathsf{y})$$

$$(b)\quad \sum_{k=1}^{n} w_k^{(i)} C_k(\mathsf{x},\mathsf{z}) < \sum_{k=1}^{n} w_k^{(i)} C_k(\mathsf{x},\mathsf{y})$$

$\overset{(iii)}{\iff}$ there exist no other surface realization z and no index i such that

$$(a)\quad C_{k_1}(\mathsf{x},\mathsf{z}) = C_{k_1}(\mathsf{x},\mathsf{y}) \quad \vdots \quad C_{k_{i-1}}(\mathsf{x},\mathsf{z}) = C_{k_{i-1}}(\mathsf{x},\mathsf{y})$$

$$(b)\quad C_{k_i}(\mathsf{x},\mathsf{z}) < C_{k_i}(\mathsf{x},\mathsf{y})$$

$\overset{(iv)}{\iff}$ there exist no other candidate z and no index i such that

(a) the top $i-1$ ranked constraints $C_{k_1},\ldots,C_{k_{i-1}}$ in (15) do not distinguish between z and y

(b) the top ith ranked constraint C_{k_i} in (15) assigns instead less violations to z than to y

6 Conclusions

The OT and HG implementations of categorical constraint-based phonology have been adopted unquestioningly throughout the last fifty years of generative phonology. This paper has argued that phonologists have been right in doing so! The gist of the argument can be summarized as follows. We consider two underlying strings such that the concatenation of any two of their candidate surface realizations is innocuous because it does not create nor dissolve relevant markedness violations. The Innocuous Concatenation Axiom (ICA) then requires any categorical grammar to realize the concatenation of any two such underlying strings as the concatenation of their surface realizations of in isolation.

The main result of this paper, boxed in section 5, provides a complete characterization of the implementations of categorical constraint-based phonology that comply with the ICA. This characterization reveals that OT and HG are the two simplest non-trivial implementations of categorical constraint-based phonology that comply with the ICA. In other words, the ICA provides an axiomatic justification of OT and HG. In the sense that OT and HG stand or fail based solely on the empirical validity of the ICA. And indeed the ICA seems empirically valid, as it has always been adopted implicitly in the daily business of phonological analysis and even hardwired into SPE grammars.

This result shows how the question "which constraint interactions?" can be disentangled from the question "which constraints?" by characterizing phonological behavior through general, formal axioms such as the ICA, that are independent of constraint substance. Finally, this result suggests that the current debate between OT (does it under-generate?) versus HG (does it over-generate?) might be misplaced. Once we realize that OT and HG are two special cases of the more general constraint-based architecture boxed in section 5, it becomes natural to try to reconcile them within the latter architecture. This project is left for future work.

References

Boersma, Paul & Joe Pater (2016). Convergence properties of a gradual learning algorithm for Harmonic Grammar. McCarthy, John & Joe Pater (eds.), *Harmonic Grammar and Harmonic Serialism*, Equinox Press, London.

Breiss, Canaan & Adam Albright (2022). Cumulative markedness effects and (non-)linearity in phonotactics. *Glossa: a journal of general linguistics* 7, 1–32.

Flemming, Edward (2021). Comparing maxent and noisy harmonic grammar. *Glossa: a journal of general linguistics* 6, 1–42.

Hayes, Bruce (2017). Varieties of Noisy Harmonic Grammar. Jesney, Karen, Charlie O'Hara, Caitlin Smith & Rachel Walker (eds.), *Proceedings of the 2016 Annual Meeting in Phonology*, Linguistic Society of America, Washington, DC.

Hayes, Bruce & Colin Wilson (2008). A Maximum Entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39, 379–440.

Magri, Giorgio (2025). Constraint interaction in probabilistic phonology: deducing MaxEnt from Hayes and Zuraw's shifted sigmoids generalization.

Magri, Giorgio & Arto Anttila (2025). The stochastic generalization and its implications for probabilistic constraint-based phonology.

McCarthy, John J. & Alan Prince (1995). Faithfulness and reduplicative identity. Beckman, Jill, Suzanne Urbanczyk & Laura Walsh Dickey (eds.), *University of Massachusetts occasional papers in linguistics 18: papers in Optimality Theory*, GLSA, Amherst, 249–384.

Pater, Joe (2009). Weighted constraints in generative linguistics. *Cognitive Science* 33, 999–1035.

Prince, Alan & Paul Smolensky (1993/2004). *Optimality Theory: Constraint Interaction in generative grammar*. Blackwell, Oxford, URL `http://roa.rutgers.edu`. Original version, Technical Report CU-CS-696-93, Department of Computer Science, University of Colorado at Boulder, and Technical Report TR-2, Rutgers Center for Cognitive Science, Rutgers University, April 1993. Available from the Rutgers Optimality Archive as ROA 537.

Riemann, Bernhard (1873). On the hypotheses which lie at the bases of geometry. *Nature* 8:183, 14–17.

Smith, Brian W. & Joe Pater (2020). French schwa and gradient cumulativity. *Glossa: a journal of general linguistics* 5, 1–33.

Smolensky, Paul & Géraldine Legendre (2006). *The Harmonic Mind*. MIT Press, Cambridge, MA.

Wilson, Colin (2025). Maximum entropy grammars. Jardine, Adam & Paul de Lacy (eds.), *The Cambridge Handbook of Phonology (2nd edition)*, Cambridge University Press.