CHAPTER II
PHONOTACTIC LEARNING AND BIASED CONSTRAINT DEMOTION

1. Introduction

The goal of this chapter is to lay out the arguments for a particular view of Optimality-Theoretic learning, instantiated in a class of learning algorithms called Biased Constraint Demotion by Prince and Tesar (2004), henceforth also known as BCD. BCD is principally a model of phonotactic learning – that is, the learner’s discovery of which marked structures are allowed in its language, in what contexts and under what circumstances.

Much of the chapter is a synthesis of the recent relevant learnability literature, drawing together work by McCarthy (1998), Smith (1999, 2000), Tesar and Smolensky (2000) and Hayes (2004), as well as Prince and Tesar. Most of the arguments are theoretically-driven, but I will also touch on supporting empirical results.

The chapter discusses in detail a proposal for adding a different bias to Biased Constraint Demotion, namely one for the most specific IO faithfulness constraint (as in Hayes’ Low-Faithfulness CD algorithm). I discuss the necessity, challenges, and workings of this bias in the second half of the chapter. The prose version of the BCD algorithm that I adopt is given in section 7.1; the reader who understands how that algorithm works need not read anything else here to understand all subsequent chapters.

1.1 Important aspects of Optimality Theory for this learning theory

This dissertation is definitely not the place for the reader to learn Optimality Theory: for introductions to the theory and its primary results, see Prince and Smolensky (1993/2004), Kager (1999) and McCarthy (2002). But I would like to make two introductory remarks about aspects of OT that make it particularly suited to the kind of acquisition questions asked in this dissertation (and that also confront it with the challenges this work tries to address.)

One important aspect of OT for our purposes is the Richness of the Base (ROTB): the strong claim that languages are defined solely on the basis of their constraint rankings, and not by their lexicons, underlying representations or indeed anything else. What this means is that although language users have very different lexicons of stored inputs, every language’s grammar can map any possible input onto a legal output – whether a real lexical item, or a nonce word judged grammatical by native speakers. The ROTB assumption places heavy demands on the phonotactic learner: the OT grammar the learner is searching for bears all responsibility for characterizing the languages’ tolerance for marked structures, because none can be shifted onto the lexicon.

Another important learning consequence of the standard OT architecture – in particular the kinds of constraints used and their range of potential conflicts -- is that many rankings can drive the same input-output mapping. This indeterminacy of ranking allowed by each input-output mapping presents the OT phonotactic learner with its main challenge: how to find the ranking that will produce the observed data and nothing more. Such a grammar is termed restrictive. As Prince and Tesar (2004) make explicit, the search for restrictiveness is the OT version of the Subset problem (Baker, 1979; Berwick,
1985; Smolensky, 1996), and it is precisely this search that the biases of Biased Constraint Demotion are built to guide.

1.2 Outline of the chapter

Section 2 introduces the main assumptions of OT learning that I adopt, taken most directly from Tesar and Smolensky (2000) and their Constraint Demotion Algorithm and emphasizing the role of stored errors in this learning model. Section 3 introduces Biased Constraint Demotion, and talks about biases that try to attribute marked structures to constraints that will allow for a maximally-restrictive grammar. I demonstrate how Prince and Tesar’s BCD algorithm works, using the well-known bias for high-ranking Markedness, and then present the argument from McCarthy (1998) for high-ranking OO-faith (see also Hayes, 2004). Section 4 introduces the trickier issue of how to attribute errors to faithfulness, focusing on the import of specific >> general relations between the contexts of faithfulness constraints, and the problems raised by Prince and Tesar in using these relations to build any simple ranking bias. Leaving aside temporarily these problems, section 5 summarizes a core success of BCD learning: that its reliance on stored errors makes it able to escape superset grammars caused by previous incorrect assumptions.

Section 6 introduces my proposal for imposing the bias for ranking specific faithfulness constraints over general ones, which relies both on constraint definitions as well as the learner’s current knowledge of the language. Section 7 synthesizes the results of sections 3 through 6 into the version of BCD that I will use in subsequent chapters; it also provides some more analysis of the ways the specificity of faithfulness constraints can be calculated and their roles in the search for restrictiveness. Finally, section 8 summarizes the chapter’s results, and considers to what extent the BCD model makes empirical predictions about the initial state of acquisition. This discussion will lead us onwards to my novel proposal for a gradual BCD learner in chapter 2.

2. Learning an Optimality-Theoretic grammar

2.1 The learning framework: the Tesar/Smolensky learner

The ways in which Optimality Theory differs from previous views of generative (rule-based) phonology also provide OT with its view of language acquisition. To learn a language-specific grammar is simply to learn a constraint ranking; to learn a language means learning that ranking, in tandem with a lexicon of underlying representations.¹

This section provides an overview of the OT learning approach used in this dissertation. For reasons of attribution as well as notational convenience, I will refer to this approach as the Tesar/Smolensky (T/S) view, with reference to their (2000) book – though as we will soon see, my implementation of this view also relies heavily on the learnability contributions of Prince, Hayes, McCarthy and others.

The T/S learner is error-driven in all of the following ways: it uses its current grammar to process ambient language data and make errors; it is the making of an error that triggers learning; and it learns by re-ranking constraints in the current grammar, guided in some way by the error.² (As we will see – these errors are just the right ones for

¹ Leaving aside the building of constraints (although see: Hayes, 1999 among others.)

² It is worth pointing out that at least one alternative proposal for learning OT rankings is error-triggered, but not really error-guided: that of Pulleyblank and Turkel (1998), (2000). Based on previous learnability work in the Principles and Parameters framework (i.e. the Genetic Algorithm approach applied to language acquisition by Clark, 1992), the Pulleyblank and Turkel learner is triggered by the errors it makes, but it selects a new ranking hypothesis through somewhat random re-combinations of rankings rather than any reasoning from the error itself.
our learner to focus on: see also Tesar and Smolensky, 2000, Prince and Tesar, 2004: 257-58.) The rest of this section will illustrate this process.

For the most part, this dissertation is concerned with learning phonotactic distributions – the possible surface structures of the language (segmental inventories and restrictions, syllable shapes, stress patterns, and the like) – and will set aside the further complicating issue of learning phonological alternations. So as a first step, we can equip our learner with the Identity Hypothesis – the hypothesis that the ambient outputs that the learner are in fact the inputs to the learner’s own grammar. Another way of stating this hypothesis is simply to invoke Lexicon Optimization (Prince and Smolensky 1993/2004 §9.3): L.O. simply says that the learner will assume an input that is identical to the output, parsed in as unmarked form as possible.

An error is an optimal candidate under the learner’s current grammar that is not identical to the observed (i.e. heard) winner. Note, however, that the observed winner will in fact match more than one possible output candidate: because although the sound signal contains all the phonetic information about the winner’s features, segments, tones, intonational contours and stress, the learner must still assign the winner its ‘hidden structure’ – feet, prosodic words, intonational phrases, morphological affiliations and the like. In the present view, learning to assign hidden structure to winners is done in tandem with the acquisition of the rest of the grammar – see the return of this point in sections 4.2-4.3 – but for now we will set it aside.

With the Identity Hypothesis in mind, we can illustrate the T/S learner. Imagine that our learner takes as input a form /A/, provides /A/ to EVAL, and receives as output the output [B]. Our current grammar has thus made an error, illustrated in the tableau in 1):

1)

<table>
<thead>
<tr>
<th></th>
<th>/A/</th>
<th>*A</th>
<th>*B</th>
<th>Ident- A vs. B</th>
<th>Ident- A vs. C</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>A</td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(ii)</td>
<td>B</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>(iii)</td>
<td>C</td>
<td></td>
<td></td>
<td>*!</td>
<td></td>
</tr>
</tbody>
</table>

Our learner’s specific task to establish why it made an error – that is, why its current grammar mapped /A/ to [B], and not to [A] – so we can ignore the rest of the candidate set and just compare the two output candidates [A] and [B]. In the version of the T/S learner that I adopt, this comparison is represented in a distilled form as in 2):

2)

<table>
<thead>
<tr>
<th></th>
<th>/A/</th>
<th>*A</th>
<th>*B</th>
<th>*C</th>
<th>Ident- A vs. B</th>
<th>Ident- A vs. C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A – B</td>
<td>L</td>
<td>W</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td></td>
</tr>
</tbody>
</table>

In Prince (2002), this distillation of candidate comparisons is called an Elementary Ranking Condition vector (Prince 2002a,b.) In this dissertation, I will refer to objects like 2) as ERC rows. What each cell in an ERC row tells us is the preference of each constraint with respect to the winner and its rival loser candidate. In this case: the tableau in 1) shows that *A assigns a violation mark to the winner [A], and no mark to the loser [B], so we can say that *A prefers the loser: thus the ERC row for the A–B comparison contains an L in the *A column. Since the second markedness constraint in the table *B assigns the opposite violation marks (one to [B] and none to [A]), *B prefers
the winner: this puts a W in its column. The third markedness constraint *C assigns equal
violation marks (in this case, none) to both the winner and loser candidates: thus, it
prefers both winner and loser equally, and this equality puts an e in the *C column.

In this way the Ls, Ws and es of an ERC row indicate the relevant discrepancies
between the current and target grammars. The core of T/S learning is to reason from these
discrepancies to necessary rankings, and the logic of OT gives us the following way to
characterize them. This logic is given as the Cancellation/Domination Lemma of Prince
and Smolensky 1993: 148; rephrased like this in Prince and Tesar 2004: 255:

3) If every L-prefering constraint is ranked below some W-prefering constraint, our
grammar will prefer the Winner to the Loser.

This lemma is the crux of the recursive Constraint Demotion Algorithm (CDA: Tesar,
1995; Tesar and Smolensky, 1998, 2000; see also Prince 2002a,b). The CDA is a
technique for modifying a ranking, by demoting Loser-prefering constraints until the
ranking no longer makes the errors encoded in its current ERC rows. We will soon see
how this works below.

This logic also drives the class of Biased Constraint Demotion algorithms (Prince
and Tesar, 2004), which I will adopt in this dissertation. In BCD, each cycle of learning
(i.e. constraint re-ranking) creates a new grammar hypothesis, and this new grammar will
cause a new set of errors and consequent mark-data pairs. While previous grammars are
forgotten as soon as a new one is built, the T/S learner I will use retains its ERC rows, in
a table called the Support. The re-ranking algorithm always works with reference to the
Support: the sum of all errors the learner has ever made.

Tesar and Smolensky (2000) provide a proof that that recursive application of the
CDA will take any Support and successfully find a ranking that ‘resolves’ all the
Support’s errors (assuming one exists.) A ranking that ‘resolves’ a set of errors is one that
chooses all the winners instead of their respective losers. In this system, the first
fundamental role of the Support is to provide the algorithm with the errors to learn from.

The Support is also crucial to tackling many subparts of the learning problem with
a BCD learner. Tesar (1997, 1998) points out that a memory for errors like the Support
allows the learner to do what he calls Inconsistency Detection, which means noticing that
no one ranking exists to describe the data. Detecting inconsistency is a specialty of the
Support, and it has many functions: see the applications of Inconsistency Detection in e.g.
Prince (2002); Tesar et al (2003); McCarthy (2005).

One application is the matter of learning “hidden structure”: properties of winners
that the sound signal does not carry, and which the learner must therefore infer. Such
structure includes both prosodic and morphological information – syllabification, footing
and higher-level prosodic structure, as well as morphological category and paradigmhood
– and also underlying representations, which may turn out to not match the observed
outputs.

As another example: the Support provides an approach to learning an OT
grammar that is sensitive to exceptions and/or lexical strata. This is the case in the
hypothesical Support given below, taken from Pater (to appear), in which every constraint
prefers some loser.
Pater (to appear) uses Inconsistency Detection to learn a grammar that encodes
exceptionality through lexically-indexed constraints. When faced with a Support like 4),
this learner finds a constraint that prefers no losers for all instances of some morpheme,
and then installs a version of that constraint indexed to all the morphemes for which it
favours only winners. (See also the discussion of exceptionality in Winslow (2003) and

The Support is also very crucial to the proposals made in this dissertation. Later in
this chapter (§6), I propose how the Support should be used to calculate contingent
ranking biases. In the next chapter, I propose a novel way by which errors get into the
Support, which derives intermediate stages of acquisition (chapter 2 §3), and also some
of the variation between those stages (chapter 2 §6.)

2.2.1 The Support and the lexicon

One question that arises in the attempt to use the T/S learner in real-life learning
is the connection between the Support and the phonological lexicon. In some ways, it
may seem that the Support should in fact be considered as a proto-lexicon, since it
contains observed words of the language that children must be learning, and since as we
will see the learner must update their entries in the Support as they learn more about e.g.
the morphological structure of those words.

Nevertheless, the Support as it stands contains both more and less information
than a phonological lexicon. On the one hand, the Support contains not just the
language’s observed outputs but also their associated losers and comparisons of violation
profiles. On the other hand, the Support only contains those forms that induced errors so
it will not include lexical items that are faithfully parseable under the current grammar.
Furthermore, the Error-Selective proposal that I make in the next chapter is very attuned
to the purely phonological properties of the errors that are added to the Support, and in
what order, with absolutely no concern for whether a child has learned the meaning or
lexical quirks of any particular error-inducing word. Thus, it seems that the Support and
the lexicon are two different mental objects, entrusted with the storage of different
knowledge; the relationship between them is an unresolved issue.

2.2.2 The Support and the Gradual Learning Algorithm

The most well-known alternative approach to learning OT is the Gradual
Learning Algorithm (e.g. Boersma, 1997; Boersma and Hayes, 2001; Boersma and
Levelt, 2000; Curtin and Zuraw, 2001; Levelt, and van der Vijver, 2004.) Although there
are other key differences between GLA learning and what I’ve discussed above – a basic
assumption of the GLA is that it has no analogous notion to the Support. Learning re-
ranks constraints gradually on the basis of one error at a time, and errors are never stored.
This means that the GLA is not equipped to handle the learning problems of hidden
structure addressed above in a consistent way.7 This difficulty with the GLA is the focus
of chapter 3§5-6.

\[ See Boersma and Appousidou (2003, 2004), where the GLA succeeds in learning metrical structure on
some trials but not others.\]
3. Restrictive phonotactic learning: Biased Constraint Demotion

Since the T/S learner is error-driven, it continues re-ranking constraints until it stops making errors. So, the crucial test of the T/S learner is that when it stops making errors, its constraint ranking is indeed the target ranking – that is, that the ranking embodies all the properties that the analyst ascribes to native adult speakers. (There is also the considerable issue of whether the learner’s inputs are also correct; which I ignore for the present although will touch briefly on in chapter 4.)

As discussed in section 1, many different constraint rankings will choose the same optimal input for a given output – which also means that each ERC row will only partially determine the nature of the new grammar to be learned. In choosing between these rankings, the OT learner runs into the classic learnability problem of subset and superset grammars, which I present below.

Given the re-ranking logic given in 3) above, the single ERC row repeated below in 5) will be resolved by any grammar that includes at least one of the rankings in 6):

5) One ERC

<table>
<thead>
<tr>
<th>input</th>
<th>winner – loser</th>
<th>*A</th>
<th>*B</th>
<th>Ident-A/B</th>
<th>Ident-A/C</th>
</tr>
</thead>
<tbody>
<tr>
<td>/A/</td>
<td>A – B</td>
<td>L</td>
<td>ε</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

6) The rankings that resolve this ERC
(a) *B >> *A
(b) Ident-A/B >> *A
(c) *C >> Ident-A/C >> *B >> *A

Clearly, many total rankings of these 5 constraints will contain one or the other of the partial rankings in 6), e.g.:

7) (a) *C >> *B >> *A >> Ident-A/B >> Ident-A/C
(b) Ident-A/B >> Ident-A/C >> *A >> *B >> *C
(c) *C >> Ident-A/C >> *B >> Ident-A/B >> *A

The crucial concern in choosing between these rankings is that our learner must not only choose a grammar to map all the /A/s to themselves instead of to [B]s. There is also the need to rule out any other unattested surface forms, [C]s through [Z]s – and as we will see shortly, some of our the rankings in (7) do this job better than others.

To use a concrete example, consider the stress rule of French, in which main stress falls on the last syllable of the word. Imagine a French learner makes the error in 8a), and so creates the ERC row in 8b):

8)a) /papá/ Trochee Lamb Non Finality Ident-Stress
(i) (pa.pá) *   *   *   *
(ii) (pápa)  *   *   *   *

8)b) input winner – loser Trochee Lamb Non Finality Ident-Stress
/papá/ (papá) → (pápa) L   W   L   W

If our learner decides to resolve this error by installing Ident-Stress above the two L-preferring constraints, the resulting grammar will be as in 9) below:

9) /papá/ Ident-Stress Trochee Lamb Non Finality
(i) (pa.pá) *   *   *   *
(ii) (pápa)  *   *   *   *
By ranking Ident-Faith highest, the learner has indeed ensured that the final stress of any input French word is preserved in the output. But it has also learned a grammar in which stress is lexically determined – merely falling wherever it is in the input. And the generative assumption is that this is NOT the grammar that French speakers have learned – it is not just a fact (or accident) of the lexicon that every single French word has ultimate stress, but also a fact we want to attribute to (and capture in) the phonological grammar of the French speaker.

From the OT perspective, we can say that our learner is searching for a grammar that faithfully reproduces all the attested forms, and also maps all of the rich base onto attested forms. In other words, we want the learner to find the most restrictive grammar.

This issue is clearly not new to OT. Much linguistic learnability work has centered on ensuring restrictiveness, and it has driven various proposals about the nature of the grammar itself.8 As we saw in the French stress example above, the error-driven OT learner suffers from the well-known subset problem because of a lack of positive evidence. The danger of relying on the current grammar to provide errors to learn from is that learners will never make errors that show they’ve chosen an insufficiently restrictive grammar. So the two goals are first to identify what makes the most restrictive grammar (constraint ranking) among any set of options consistent with the data, and then to ensure that the learner learns that ranking.

As already cited, the search for restrictiveness in OT learning is at the core of the proposals in Prince and Tesar (2004) and also Hayes (2004) – in what follows, I start from the Prince and Tesar model of BCD, but I adopt insights and technology from both works. The central idea of BCD is to give the learner a set of prior assumptions about constraint rankings, called ranking biases, which the learner assumes up until the learning data provides evidence to the contrary. Building a constraint ranking is a series of cycles of adding constraints to strata – starting at the top and continuing until there are no more constraints to be ranked. In building each stratum, the learner aims to install all constraints that its biases want highest-ranked, and put off the installation of all other constraints until it has to.

To see how Biased Constraint Demotion works, I will start with an illustration of the most basic for choosing the most restrictive OT grammar: M >> F.

3.1 Illustrating the BCD approach: high-ranking Markedness

In the terms of OT learnability, the Markedness >> Faithfulness bias first appears in Smolensky (1996) and Tesar and Smolensky (1998) (elaborating on a suggestion made by Alan Prince.) In the literature on children’s productions, this observation goes back at least to Jakobson (1941/1968); see also the works of e.g. Jakobson and Halle (1956), Stampe (1969), Macken (1978), Dinnsen (1992); Fikkert (1994); and in the OT context, Gnanadesikan (1995), Demuth (1995), Pater (1997). The more Markedness is high and Faithfulness is low in a grammar, the fewer marked surface structures it permits in the language, and thus the more restrictive the grammar is. So if the constraint that the learner is going to use to choose a loser over a winner could either be an M or an F constraint, the drive for restrictiveness should make them choose the M constraint.

Let us see how the BCD approach enforces M >> F. Note that while we are only using one bias here, the same reasoning will apply no matter how many biases we add.

---

8 See e.g. Dresher and Kaye (1990); Dresher (1999).
into the model. In this example, we will assume our learner has only added one error to the Support, being the schematic error we’ve already seen:

10) the Support – a collection of ERCs

<table>
<thead>
<tr>
<th>input</th>
<th>winner – loser</th>
<th>*A</th>
<th>*C</th>
<th>*B</th>
<th>Ident-A/B</th>
<th>Ident-A/C</th>
</tr>
</thead>
<tbody>
<tr>
<td>/A/</td>
<td>A ~ B</td>
<td>L</td>
<td>2</td>
<td>W</td>
<td>W</td>
<td>2</td>
</tr>
</tbody>
</table>

What we’ve already seen is that any ranking with either *B or Ident-A/B >> *A will get the winner to be more harmonic than the loser; what we know now is that we want to choose the *B >> *A ranking. And we now have two competing goals: resolving the error in 10) and respecting the M >> F ranking bias.

Remembering the logic of CDA: resolving errors means installing some W-prefering constraint over every L-prefering constraint. Once this has been done for any particular ERC row, its loser is guaranteed to be less optimal than its winner, and so that error can be ignored for the rest of the ranking process. So the BCD imposes the M >> F bias by first installing all M constraints that do not prefer the loser, and then checking whether the error has been resolved:

11) Step 1: Install all M constraints that prefer no losers
Resulting stratum 1: *B, *C

Looking back at our MDP in 10), we can see that our error has indeed been resolved, because one of the installed constraints, *B, prefers the winner, so:

12) Remove from consideration all resolved errors
Resulting Support: -- empty --

Now we go through the second cycle, to add constraints into stratum 2. Since there are no remaining errors, there are no constraints to prefer any losers, so our bias is free to install all the remaining markedness constraints:

13) Step 1: Install all M constraints that prefer no losers
Resulting stratum 2: *A

14) Full ranking so far: *B, *C >> *A

(This also means we have no errors to remove in part 2.)

Now we have installed all the constraints that our bias wants to rank high. This means we are safe to dump all the remaining constraints (the faithfulness constraints) at the bottom of the hierarchy:

15) Step 1: Install all M constraints that prefer no losers
Resulting stratum 3: -- empty --

Step 2: Install all remaining constraints in the last stratum (to be revised)*
Resulting stratum 3: Ident A/B, Ident A/C

16) Final full ranking: *B, *C >> *A >> Ident A/B, Ident A/C

Happily, BCD has found the right ranking: the ranking in 15) chooses the winner [A] over its rival [B]:

---


---

9 The discussion of how faithfulness constraints should be installed, even when errors still remain, will be extensive – see §4.6-7 below.
17)a) The correctness of $B \gg A$

<table>
<thead>
<tr>
<th></th>
<th>*A</th>
<th>*B</th>
<th>*C</th>
<th>Ident</th>
<th>Ident</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>A</td>
<td>-</td>
<td>-</td>
<td>A/B</td>
<td>A/C</td>
</tr>
<tr>
<td>(ii)</td>
<td>B</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

And because we chose the $B \gg A$ ranking to get our winner to beat its rival loser, the learner has not made any unmotivated concessions to faithfulness. So, for example, if we were now to encounter an input that has features protected by Ident A/B, it will still be neutralized to something less marked.

17)b) The restrictiveness of $B \gg A$

<table>
<thead>
<tr>
<th></th>
<th>*A</th>
<th>*B</th>
<th>*C</th>
<th>Ident</th>
<th>Ident</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>A</td>
<td>-</td>
<td>-</td>
<td>A/B</td>
<td>A/C</td>
</tr>
<tr>
<td>(ii)</td>
<td>B</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

A note: it will already be clear that BCD does not build completely classic OT grammars in one sense, because the ranking in 15) is a stratified, partial ordering of constraints rather than a total ordering – but any total constraint ordering that is consistent with this partial ordering can be selected.

Stepping back: we can say that the role of the $M \gg F$ bias is to put as little burden on the lexicon and as much on the grammar as possible. In other words, it chooses rankings that will rule out as many unseen forms as possible, without requiring negative evidence of their non-existence. Of course, it will soon come to pass that a learner’s positive evidence cannot be resolved by installing Markedness constraints alone. The second ranking bias which I integrate into Biased Constraint Demotion provides some additional help: this is the bias for high-ranking OO-Faith.

3.2 A second bias in BCD terms: high-ranking OO-Faith

Although the central notion of faithfulness of Prince and Smolensky (1993/2004) is between input and output (IO-faith), work in OT since then has adopted a variety of different faithfulness relations – e.g. between bases and reduplicants (McCarthy and Prince, 1995 et seq.), and also between morphologically-simple and derived outputs. Here I will focus on this latter relation.

As the name suggests, OO-faith constraints assess similarity among output forms. Output-output relations are the OT answer to the long-observed phenomenon of paradigm uniformity: i.e., that the phonological regularities of a language are often overridden just where they would cause morphologically-derived forms to differ from their bases. In other words: some phonological generalizations only have exceptions that keep the derived forms of a morphological paradigm similar to their base. In the spirit of this wording, the OT accounts of paradigm uniformity (defined variously: see Burzio 1997, 2000; Kenstowicz, 1997; Kager, 1998; Steriade 1998, 2000 inter alia) all enforce something akin to faithfulness between morphologically-related surface forms. The choice of proposals is not crucial -- as far as I know, the learnability arguments to follow do not hinge on any particular account of paradigm uniformity.
3.2.1 **OO faith as an OT account of cyclicity**

One famous example of the phenomenon is the interaction of flapping and Canadian raising (CR) in some dialects of English (e.g. Joos 1942; Chambers 1973; Mielke et al 2003). In such dialects, CR is purely allophonic in monomorphemic words: raised [a] appears before voiceless obstruents as in ‘write’ [waɪt], while [a] appears elsewhere as in ‘ride’ [raɪd]. However, derived forms with a base vowel [a] exceptionally retain their raised quality even before a voiced flap, as in ‘writer’ [waɪtə], *[waɪtə]. In other words: diphthongs are unraised before flaps except when they are raised in the base.12

11 By ‘some dialects’, I refer to one of the two dialects originally reported in Joos (1942). Whether this particular dialect is anything more than an idealization among modern-day speakers is a separate question: see especially Hall (2005).

12 In fact, it appears that the intervocalic flapping process that creates the environment for exception raising in ‘writer’ is in fact also OO-Faith sensitive in longer words -- see Wittgott, 1982; Steriade, 2000; Davis 2002 on ‘cap[i]alist’ vs.’mil[i]aristic’.

18 **Exceptional Canadian Raising in words derived from raised bases**

rider, [ˈraɪdər] vs. writer, [ˈwaɪtə] (c.f. ‘write’, [ˈwaɪt])

wider, [ˈwaɪdər] whiter, [ˈwaɪθə] (c.f. ‘white’ [ˈwaɪθ])

The constraint set I will use here: **Output-Output Faithfulness** (Benua 1997, 2000)

19 **Output-Output-Faith-[F], informally defined**

“Derived words must match their base’s value for the feature [F]”

The schematic analysis of such a pattern will be OO-Faith >> Mark >> IO-Faith. First, Mark >> IO-Faith ensures the normal distribution on raised diphthongs; in this environment (i.e. before a voiced flap) the context-free markedness constraint against raised diphthongs rules [a] out (see 21a below). OO-faith >> M enforces exceptional raising: an undominated OO-faithfulness constraint that regulates vowel height (OO-Ident-[hi]) will protect raised diphthongs only to keep derived forms similar to their bases (21b):

20 **The constraints**

* A1

OO-Ident-[hi] Vowels in derived outputs must match their output bases correspondent’s value for the feature [hi]

IO-Ident-[hi] Vowels in outputs must match their input correspondent’s value for the feature [hi]

21 **The rankings**

(a)  (b)

<table>
<thead>
<tr>
<th>*A1</th>
<th>IO-Ident-[hi]</th>
<th>*A1</th>
<th>IO-Ident-[hi]</th>
</tr>
</thead>
<tbody>
<tr>
<td>rider</td>
<td>*A1</td>
<td>*(JAT)</td>
<td>*(JAT)</td>
</tr>
<tr>
<td>wider</td>
<td>*(JAT)</td>
<td>*</td>
<td>*(JAT)</td>
</tr>
</tbody>
</table>

Examples of this phenomena are robustly attested across languages – famously, stress in derived words is often constrained by paradigm uniformity (on Arabic, see e.g. Brane, 1974, Kager, 1999; on English stress, see e.g. Chomsky and Halle, 1968; Pater, 2000.) Two other, different examples are illustrated below:

22) English sonorants are syllabified as onsets before vowels except when they are syllabified as nuclei in the base:

<table>
<thead>
<tr>
<th>Word</th>
<th>Syllabification</th>
</tr>
</thead>
<tbody>
<tr>
<td>light.ing</td>
<td>onset</td>
</tr>
<tr>
<td>ligh.ten</td>
<td>onset</td>
</tr>
<tr>
<td>William Faulk.ner</td>
<td>onset</td>
</tr>
<tr>
<td>Hugh Heff.ner</td>
<td>onset</td>
</tr>
</tbody>
</table>

23) In Sundanese, nasalization does not spread across an oral consonant except when the target of spreading is nasalized in the base (Robins, 1957; Cohn, 1990; Walker, 1998)

(a) [+nasal] spreads rightwards only through vowels/glottals

<table>
<thead>
<tr>
<th>Word</th>
<th>Syllabification</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ni:]</td>
<td>‘seek’</td>
</tr>
<tr>
<td>[niiha:]</td>
<td>‘expensive’</td>
</tr>
<tr>
<td>[nighia]</td>
<td>‘to be rich’</td>
</tr>
</tbody>
</table>

(b) spreading blocked by [r] and [l]

<table>
<thead>
<tr>
<th>Word</th>
<th>Syllabification</th>
</tr>
</thead>
<tbody>
<tr>
<td>[bilia:]</td>
<td>‘stretch’</td>
</tr>
<tr>
<td>[bitarios]</td>
<td>‘examine’</td>
</tr>
</tbody>
</table>

(c) but base vowels still nasalized even across the infix [-al] / [-ar]

<table>
<thead>
<tr>
<th>Word</th>
<th>Syllabification</th>
</tr>
</thead>
<tbody>
<tr>
<td>[al-]:</td>
<td>‘seek, plural’</td>
</tr>
<tr>
<td>[al-ar]:</td>
<td>‘expensive, plural’</td>
</tr>
</tbody>
</table>

3.2.2 OO-faith as an OT account of MSCs

A different use of OO-faith, relevant to the learning discussion to follow, is McCarthy (1998)’s OT reanalysis of Morpheme Structure Constraints (see e.g. Chomsky and Halle 1968; Kisseberth 1970.) McCarthy uses the example of the distribution of root vowel length and Minimal Words in the ‘Kansai B’ dialect of Japanese. In this dialect, there is a static generalization that roots are always at least bimoraic (e.g. [kaa], *[ka]) – this is true independent of any surrounding morphology.

24) Kansai /B/

possible paradigms: impossible paradigms:

<table>
<thead>
<tr>
<th>Root</th>
<th>Root + Affix</th>
<th>Root</th>
<th>Root + Affix</th>
</tr>
</thead>
<tbody>
<tr>
<td>[kaa]</td>
<td>[kaaga]</td>
<td>*[ka]</td>
<td>[kaga]</td>
</tr>
<tr>
<td>[kaga]</td>
<td></td>
<td>*[kaga]</td>
<td></td>
</tr>
</tbody>
</table>

Assuming that the bimoraicity minimum is the work of the high-ranking markedness constraint FootBinarity (Prince and Smolensky, 1993), we can explain why paradigms like (24b) do not occur, because FtBin >> IO faithfulness to syllable weight will force any input mono-moraic root to lengthen on the surface:

25) /ka/ FtBin IO-Ident-Wt

<table>
<thead>
<tr>
<th>Surface</th>
<th>FtBin</th>
<th>IO-Ident-Wt</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) ka</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>(ii) kaa</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

But if we can have input roots like /ka/, why are there no surface paradigms like (24c)? That is: what forces a root to remain lengthened, even in derived forms where word minimality is no longer at issue? McCarthy’s point is that paradigms that alternate can be ruled out with OO-faith to syllable weight (i.e. moras). A grammar in which OO-Ident-Wt outranks its IO-faith counterpart will give us this result:

26) /ka + ga/ Ft-Bin OO-Ident-Wt IO-Ident-Wt

<table>
<thead>
<tr>
<th>Surface</th>
<th>Ft-Bin</th>
<th>OO-Ident-Wt</th>
<th>IO-Ident-Wt</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) kaga</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>(ii) kaaga</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

The role of OO-faith here is to ensure that derived forms match their bases for vowel length -- even if Markedness does not require long vowels in that environment.
3.2.3 The learnability argument for high-ranking OO-faith: McCarthy (1998)

The learnability argument for high-ranking OO-Faith bias also comes from McCarthy’s discussion above. According to the ranking in (26), Markedness ensures that roots are bimoraic in simple words, while OO-Faith insists that the root portion of a complex word remain similarly bimoraic. How can the learner get to this ranking? While the M >> F bias means that the initial state already contains the ranking that lengthens hypothetical roots in 25), nothing in the data will drive the ranking between OO- and IO-Faith necessary in 26).

To restate the argument in the terms of our BCD model: this learner of the non-alternating dialect will only have evidence for inputs like /kaa/ and /kaaga/, and so only have the two ERCs in 27) (note that I have included the M constraint *LongV in this Support table, to explain why the learner might makes these errors in the first place):

<table>
<thead>
<tr>
<th>input</th>
<th>winner – loser</th>
<th>FtBin</th>
<th>*Long-V</th>
<th>OO-Ident-Wt</th>
<th>IO-Ident-Wt</th>
</tr>
</thead>
<tbody>
<tr>
<td>/kaa/</td>
<td>kaa – ka</td>
<td>W</td>
<td>L</td>
<td>e</td>
<td>W</td>
</tr>
<tr>
<td>/kaa + ga/</td>
<td>kaaga – kaga</td>
<td>e</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

This Support merely tells the BCD learner that *LongV needs to be demoted below some winner-prefering constraints – and that given the error in the derived context, *LongV must get below either OO or IO faith. McCarthy’s point is thus that a ranking bias for OO >> IO faith is necessary. This bias will ensure that OO-Ident[hi] is installed high and therefore exclude the possibility of alternating paradigms like (24c).

This example provides the restrictiveness argument for ranking OO-faith over IO-faith; it says nothing about its interaction with Markedness. In the version of BCD I will put together at the end of this chapter (§7.1), however, OO-faith will in fact be installed above Markedness constraints wherever possible as well. The arguments I will provide for this choice will come from empirical evidence from developmental stages in the literature, noted by Hayes (2004) and others – this data will be discussed in chapter 4 7.3.1. However, one can also demonstrate that an OO-Faith >> Markedness bias is required to ensure the acquisition of some end-state grammars as well: see Becker (2006).

3.3 Connecting M >> F and OO >> IO as surface-oriented biases

Bruce Tesar (p.c.) points out that the high-ranking M and OO-F biases have in common a reliance on surface evidence. In other words: the violation profiles of M and OO-faith are defined exclusively by looking at the outputs that children have not just hypothesized, but actually heard. As a result, their full potential for responsibility for surface contrasts and neutralizations is already known.14

One related point is that this surface-oriented property makes it straightforward to learn the appropriate context of markedness and OO-faith activity using error-driven learning. To see this, I show here how our M >> F biased learner correctly acquires a grammars in which a specific and general markedness constraint are crucially ranked with respect to one another.

This very simple example is the case of a language in which coda consonants are allowed, but coda clusters are not permitted (Blevins 1995 lists the languages Thargari, Sedang and Mokilese as having this syllable profile.) The analysis of this pattern that I will adopt in chapter 2 is one with two markedness constraints in a stringency relation:

14 The one hypothesis that the learner must make in the case of OO-faith is to determine which segments in a derived form make up the morphological base.
general NoCoda constraint, and a more specific NoComplexCoda constraint. The singleton coda pattern comes from sandwiching faithfulness between these markedness constraints. If we assume for example that coda clusters would be repaired via deletion, we can use the ranking in 28a) below to derive the right results:

28) The singleton coda grammar
(a) NoComplexCoda >> Max >> NoCoda
(b) Faith >> General M
(c) Specific-M >> Faith

<table>
<thead>
<tr>
<th>/bab/</th>
<th>Max</th>
<th>NoCoda</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>bab</td>
<td>*</td>
</tr>
<tr>
<td>(ii)</td>
<td>ba</td>
<td>*!</td>
</tr>
</tbody>
</table>

The error-driven BCD learner will discover the correct ranking in 28a) without incident. Our phonotactic learner will only make one kind of error – one which reduces the singleton codas it hears in words like [bab] – and this error will produce the ERC in 29) below:

29) ERCs for the singleton coda grammar

<table>
<thead>
<tr>
<th>winner – loser</th>
<th>*ComplexCoda</th>
<th>NoCoda</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>bab – ba</td>
<td>e</td>
<td>L</td>
<td>W</td>
</tr>
</tbody>
</table>

On the first pass of learning from this error, the high-ranking M bias from section 3.1 will first install the specific *ComplexCoda in the top stratum, simply because it does not prefer the loser. Since it does not prefer the winner either, though, it will not resolve the error, so BCD will be forced to install faithfulness in the second stratum and then place NoCoda at the bottom of the ranking. This process generates the correct ranking from 29a).


What is perhaps less obvious is that this ease of learning holds of OO-faith as well. That is, if various OO-faith constraints with different contexts can explain the surface structure of derived forms, the learner’s errors will demonstrate which ones should be demoted. The following illustration comes from the famous example of cyclic effects in Palestinian Arabic (Brame 1974), in which just those vowels with main stress in a morphological base escape vowel syncope in derived forms: the data are summarized in 30) below.

30) a) In Palestinian Arabic, unstressed [i] is usually deleted:
   /fihim-u/obj → [fîhmu], *[fîhimu]
   /fihim-na/obj → [fîhimna], *[fîhimna]

b) even if the [i] comes from a morphological base (bases underlined):
   /fihim/ → [fîhim]
   and: /fihim-u/sec → [fîhmu], *[fîhimu]

c) … except when the base [i] was stressed:
   /fihim/ → [fîhim] so: /fihim-na/sec → [fîhimna], *[fîhimna]

This effect is re-analyzed by Kager (1999)\(^{15}\) using OO-Max constraints relativized to segments in the base’s prosodic head (see also McCarthy, 1995a; Alderete, 1999). The relevant constraint, in both Kager’s definition and the current OO-faith terms, is in 31):

31) Head-Max(B/O): ‘Every segment in the base prosodic head has a correspondent in the output’ (Kager, 1999: 214)
   in other words: OO-Max[Seg]-prosodic head

\(^{15}\) See also Kenstowicz, 1997; Steriade, 1998.
In comparing the base and derived forms in 30b) and c), the learner gets the overt evidence that OO-Max-Prosodic Head is satisfied while general OO-Max is violated (as well as IO-Max):

32)  An OO-faithful grammar

(a)  M >> General-OO-faith

<table>
<thead>
<tr>
<th>/fíhim-ũ/acc</th>
<th>Syncope</th>
<th>OO-Max[Seg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>/fíhimũ̂/base</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(i)  /fíhimũ̂/base ~ /fíhimũ̂/base

(ii)  /fíhimũ̂/base ~ /fíhimũ̂/base

(b)  Specific-OO-faith >> M >> General-OO-faith

<table>
<thead>
<tr>
<th>/fíhim-ňa/acc</th>
<th>OO-Max[Seg]</th>
<th>Syncope</th>
<th>OO-Max[Seg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>/fíhimũ̂/base</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(i)  /fíhimũ̂/base ~ /fíhimũ̂/base

(ii)  /fíhimũ̂/base ~ /fíhimũ̂/base

With these violation profiles, the learner does not need any further bias apart from its preference for OO-faith constraints, and for constraints that prefer no losers, to get the right ranking. Since we are assuming an initial grammar where OO-faith is undominated, the learner will only be making errors like 33a):

33)  ERC row for the Palestinian Arabic syncope grammar

<table>
<thead>
<tr>
<th>/fíhim-ũ/acc</th>
<th>Syncope</th>
<th>OO-Max[Seg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>/fíhimũ̂/base</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This error makes it clear that Syncope must rank above the general OO-Max constraint; this will produce the ranking 32b).
4.1 The theory of positional faithfulness

The positional view of faithfulness constraints that I adopt takes as its starting point a general observation that is so well summed up in Smith (2002) that I quote verbatim:

34) “There is a set of phonologically prominent or "strong" positions that are well known for their special ability to license phonological contrasts, resisting neutralization processes that may otherwise be active in a language (Trubetzkoy 1939; Steriade 1993, 1995, 1997; Beckman 1995, 1997, 1998; Casali, 1997; Padgett 1995; Lombardi 1990; Zoll 1996, 1997, 1998) […] Many languages will tolerate a particular phonological contrast, such as that between voiced and voiceless obstruents or that between oral and nasal vowels, only inside one of these strong positions. Specific examples of special contrast-licensing behavior in the various strong positions can be found in the references cited above. (Smith 2002: 8).”


Positional faithfulness constraints are thus an encapsulation of this “special ability to license phonological contrasts”: they formalize the claim that it’s not an accident that some languages contrast voicing in onset and neutralize it in coda, but not the other way around. A fairly comprehensive set of such positions are used to define constraints in 35) below – here I use them to create a series of Ident[voice] constraints:

35) A set of Ident[voice] constraints

a) Ident[voice]-segment  “Output segments must match their input correspondents for the feature [voice]”

b) Ident[voice]-V:  “Output long vowel segments must match their input correspondents for the feature [voice]”

c) Ident[voice]-Onset  

“Output segments in syllable onsets must match their input correspondents for the feature [voice]”

d) Ident[voice]-σ1  “Output segments in initial syllables must match their input correspondents for the feature [voice]”

e) Ident[voice]-σ  “Output segments in stressed syllables must match their input correspondents for the feature [voice]”

f) Ident[voice]-Root  “Output segments in morphological roots must match their input correspondents for the feature [voice]”

g) Ident[voice]-Noun  “Output segments in nouns must match their input correspondents for the feature [voice]”

These Ident constraints have been defined with respect to output contexts. As we will see in chapter 2, other contextual faithfulness constraints – at least positional Max -- must be differently defined, either by referring to input contexts or through some other mechanism. These definitional issues are not the focus of this work, but I will point out the definitional assumptions necessary to my analyses when they arise.

4.1.1 Why not (only) positional markedness

This dissertation does not argue that positional faithfulness constraints are the only way the grammar encodes contextual sensitivities – that is, that there are no positional markedness constraints. However, neither do I adopt the position of Prince and Tesar (2004) that positional markedness constraints should be the only way, in virtue of their ease in learning. Instead, I claim that some positional faithfulness constraints are

16 It has been suggested that the proper context of this constraint is ‘released consonants’ or something similarly phonetic in its definition (see e.g. Kingston, 1985; Steriade, 1999; Côté, 2000.) In this work, however, I will continue to use the syllabic position Onset.
indeed necessary to capture the range of both developing and adult grammars, and therefore that their learning consequences must be taken seriously.

To support this claim, this section puts the learnability arguments of this chapter on hold and provides three arguments in favour of positional faithfulness. These arguments are (i) its ability to capture the similarity between positional neutralization and assimilation, (ii) its ability to characterize strong positions as blockers, as pointed out in Beckman (1998) and (iii) its ability to capture generalizations about positions whose complements seem to be non-categories.

The first argument comes to me from Pater (p.c.); it originates in part in Mester and Ito (1989)'s analysis of onset-driven voicing assimilation, and was taken up in the OT literature by Cho (1990) and Lombardi (1991, 1996, 1999). The relevant generalization is that onsets both preferentially resist obstruent voicing neutralization as compared to codas, and also preferentially determine the value of coda-onset voicing assimilation. As cited to this end by Lombardi (1996), languages like Polish, Dutch, Catalan and Sanskrit demonstrate this privilege of onset voicing, in that the voice specification of their obstruent clusters is determined by the input voicing of the onset segment and their word-final segments are uniformly voiceless. As 36) illustrates, the positional faithfulness constraint Ident-Onset provides a unified account of both cross-linguistic tendencies:

\[\text{Ident-Onset}^{\text{onset}} \rightarrow \text{voice specification}\]

On the other hand, a positional markedness constraint like *Coda-VoicedObstruent does not provide the same connection. While this constraint can also explains contextual neutralization as in 37a) below, it cannot explain why a language would ever resolve a coda-onset mismatch by becoming uniformly voiced:

37) The positional markedness account:

a) coda (word-final) neutralization

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{ident} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{bad} & & & \\
\hline
\text{bad} & & & \\
\hline
\text{pat} & & & \\
\hline
\end{array}
\]

b) onset-driven cluster assimilation

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{agreement} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{atpa} & & & \\
\hline
\text{adpa} & & & \\
\hline
\text{atba} & & & \\
\hline
\end{array}
\]

\[\text{agreement}^{\text{onset}} \rightarrow \text{voice specification}\]

(c) ... to either voice value

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{ident} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{atpa} & & & \\
\hline
\text{atba} & & & \\
\hline
\text{adpa} & & & \\
\hline
\end{array}
\]

36) The positional faithfulness account

a) coda (word-final) neutralization

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{ident} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{bad} & & & \\
\hline
\text{bad} & & & \\
\hline
\text{pat} & & & \\
\hline
\end{array}
\]

b) onset-driven cluster assimilation

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{agreement} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{atpa} & & & \\
\hline
\text{adpa} & & & \\
\hline
\text{atba} & & & \\
\hline
\end{array}
\]

(c) ... to either voice value

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{ident} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{atpa} & & & \\
\hline
\text{atba} & & & \\
\hline
\text{adpa} & & & \\
\hline
\end{array}
\]

\[\text{agreement}^{\text{onset}} \rightarrow \text{voice specification}\]

\[\text{agreement}^{\text{coda}} \rightarrow \text{voice specification}\]

On the other hand, a positional markedness constraint like *Coda-VoicedObstruent does not provide the same connection. While this constraint can also explains contextual neutralization as in 37a) below, it cannot explain why a language would ever resolve a coda-onset mismatch by becoming uniformly voiced:

37) The positional markedness account:

a) coda (word-final) neutralization

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{ident} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{bad} & & & \\
\hline
\text{bad} & & & \\
\hline
\text{pat} & & & \\
\hline
\end{array}
\]

b) ... but not coda-to-onset assimilation (cf. 36c with winner [adba])

\[
\begin{array}{|c|c|c|c|}
\hline
\text{word} & \text{agreement} & \text{coda-neutralization} & \text{onset-neutralization} \\
\hline
\text{atpa} & & & \\
\hline
\text{atba} & & & \\
\hline
\text{adpa} & & & \\
\hline
\end{array}
\]

\[\text{agreement}^{\text{coda}} \rightarrow \text{voice specification}\]

\[\text{agreement}^{\text{onset}} \rightarrow \text{voice specification}\]
This argument also demonstrates the advantage of faithfulness constraints relativized to featural contexts as well. Similar to the onset/coda case above, it is also the case that stops both resist place neutralization over nasals, and also determine the direction of nasal/stop place assimilation (Joe Pater, p.c.). See also especially Steriade (2000) on the special behaviour of retroflex consonants in both direction of assimilation and positional neutralization compared to other places of articulation. I will return to featurally-limited faithfulness of this sort in §6.1.

The second argument is the ability of segments in privileged contexts to block phonological processes, in a way that cannot be expressed in terms of markedness. The compelling example of the blocking comes from Beckman’s discussion of Guaraní vowel harmony (Beckman 1998: 153-184.) The relevant distribution of Guaraní nasality can be described as follows: (a) stressed vowels freely contrast for nasality (they are either nasal or oral), and (b) unstressed vowels and sonorants are only nasalized through a process of nasal assimilation, in which nasality spreads leftward from a stressed nasal vowel, nasalizing all sonorants and passing transparently through voiceless obstruents, “up to but not including the next stressed vowel” (Beckman 1998: 157, her emphasis.) The examples below demonstrate how this long-distance nasal harmony is stopped by a stressed vowel, whether nasal or oral (note that due to the typographic messiness of marking nasality and stress above vowels, I have marked nasality by underlining each nasalized segment):

To capture the fact that only stressed vowels are freely nasalized, the positional faithfulness story uses the ranking Ident[nasal]-σ' >> *NasalV >> Ident[nasal]-Seg:

The positional faithfulness constraint for stressed vowels will also capture the fact that stressed oral vowels block the continued spread of nasality in data like (38). Whatever constraint drives nasal harmony (here I simply adopt Beckman’s use of Align-L[nasal]), this markedness constraint is also ranked between the two Ident[nasal] constraints and so can be violated only to preserve the nasality of a stressed vowel (compare 40ii and iii):

18Beckman (footnote 10) tells us that the morpheme given in the input of this example as /ro/ with an unstressed nasal vowel is in fact a reduced version of the conjunction/postposition /ramó/ seen in the previous example, whose nasality is expected. There is also some rightward nasal spreading from /ro/’s unstressed vowel; see Beckman’s footnote 9 on the phonological status and treatment of progressive nasal harmony in Guaraní, which does not concern us here.
The aspect of Guaraní that argues crucially for a positional faithfulness account is this pattern of blocking in (40) above. The positional markedness alternative to account for Guaraní would replace the constraint $\text{Ident[nasal]-σ'}$ with a constraint like $\text{License[nasal]}$ – a markedness constraint that requires nasal vowels to be associated with strong positions, such as the stressed syllable (see Flemming 1993; Steriade, 1995 for a fuller story.) The problem for this account is how to handle the mapping in 40) – why should stressed oral vowels block the spread of nasality? As Beckman puts it, the claim of $\text{License[nasal]}$ is that “[+nasal] is licensed whenever it is associated to a stressed syllable, regardless of its input source. The underlying nasality/orality of the stressed vowel is irrelevant” (Beckman 1998: 183.) And as 41) below shows, no matter how these markedness constraints are re-ranked they cannot explain why nasal harmony does not spread through a stressed vowel that is underlingly oral:

41) Failure to block nasal harmony with $\text{License[nasal]}$ (from Beckman 1998: 183)

<table>
<thead>
<tr>
<th>Input</th>
<th>License[nasal]</th>
<th>Align-L(nasal)</th>
<th>$\text{Ident[nasal]-Seg}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) rexótaramó</td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(ii) rexótaramó</td>
<td></td>
<td></td>
<td>**</td>
</tr>
<tr>
<td>(iii) rexótaramó</td>
<td></td>
<td></td>
<td>**</td>
</tr>
</tbody>
</table>

This ability to block processes that would change the input specifications of strong positions is thus another reason to maintain the existence of specific faithfulness in the grammar.

The third argument about the best definition of contexts is exemplified by the proposal of Noun-Faithfulness in Smith (2001). Smith demonstrates that a variety of languages display a wider set of contrasts in nouns than other categories (verbs, adjectives, function words, etc.) One of Smith’s examples comes from the accentual system of Fukuoka dialects of Japanese, in which nouns escape the otherwise-general pattern of pitch accents on the penultimate mora of lexical items:

42) Pitch accent in Fukuoka Japanese:

(a) Verbs/adjectives: pitch accent on $σ$ with penultimate mora:

\[
\begin{align*}
\text{tabé} &\quad \text{‘to eat’} & \text{akáka} &\quad \text{‘red’} \\
\text{tabé} &\quad \text{‘to eat’} & \text{akákaró} &\quad \text{‘probably red’}
\end{align*}
\]

(b) Nouns: faithful to pitch accent on other moras,

\[
\begin{align*}
\text{inói} &\quad \text{‘life’} & \text{óokami} &\quad \text{‘wolf’} \\
\text{atama} &\quad \text{‘head’} & \text{unaccented}
\end{align*}
\]

… and to lack of accent on the penultimate mora:

Smith’s point includes the fact that this special property of nouns appears to be fairly asymmetric: that languages like Fukuoka-prime in which pitch accent is predictably assigned in all morphological categories except verbs, except adjectives, etc. are unattested. Thus, positing markedness constraints relativized to every lexical class except nouns seems to suggest we’re missing something: i.e., faithfulness to nouns.

With these arguments in hand, we will now return to the learning discussion under the assumption that positional faithfulness constraints form part of CON, and so must form part of our learnability story.
4.1.2 Stringency, not fixed rankings

To use the terminology of Prince (1997) and de Lacy (2002): the sets of faithfulness constraints that I have adopted in 35) above stand in *stringency* relations. The constraint Ident-Onset[F] is *less stringent* than Ident-Segment[F] because the former assigns a proper subset of the violation marks assigned by the latter; in other words, violating a less stringent constraint entails violating a more stringent one.

The effects of stringency have also been derived using fixed rankings of *two sets* of specific constraints. For example, McCarthy and Prince (1995)’s approach to the phonological privilege of roots is to split faithfulness into Root-Faith and Affix-Faith versions, and to propose the ‘meta-ranking’ (fixed ranking) of Root >> Affix.

Throughout this dissertation, I will be adopting stringency rather than fixed rankings. For starters, we will see below in section 4.3 that stringency relations between faithfulness constraints can be *language-specific*, and therefore uncapturable in any fixed ranking. And though they are often similar in their effects, the fixed-ranking and stringency approaches also do make different typological predictions. These differences stem from the effects of what Prince terms an ‘Anti-Paninian’ ranking.\(^\text{19}\) An Anti-Paninian ranking is a crucial ranking of a less stringent constraint above more stringent one (i.e., General-F >> Specific-F) – and the effects of Anti-Paninian rankings cannot be replicated in a fixed ranking model, because they are precisely what the model prevents.

While the need for Anti-Paninian rankings between markedness constraints, and therefore the use of stringent definitions, seems fairly solid (see especially de Lacy (2002)’s extensive cross-linguistic discussion of sonority-driven stress) the choice between fixed rankings and stringency for faithfulness is less clear. Several proposals

---

19 See the Prince and Smolensky (1993) appendix on Panini’s theorem, which gives rise to the term.


have been made which rely on General-Faith >> Specific-Faith rankings to produce attested patterns: see Keer (1999: 82-85) on Fula geminate hardening; Lombardi (1999) on Swedish voicing assimilation; de Lacy (2002: chapter 8) on Chipewyan coalescence and other patterns.\(^\text{20}\) However these analyses suffer from a typologically-uncomfortable prediction known as ‘Majority Rules’ (see Bakovic, 1999ab; Lombardi, 1999; Wilson, 2000; de Lacy 2002: §7.7.3) -- a problem whose real scope and possible solution I will not address here. I will return to Anti-Paninian rankings and phonotactic learning in §7.3.

4.2 The learnability argument for a Specific-F >> General-F bias: Smith (2000)

In the same spirit as the high-ranking M bias: we want our learner to assume rankings that resolve errors while being IO-faithful in as few contexts as possible. This means that when choosing a ranking that is faithful to the input, the learner should only install the *least stringent* (i.e. *most specific*) F constraint that can resolve an error. This bias aims at avoiding the learning of superset grammars, as spelled out in Smith (1999, 2000) as well as in Hayes (2004).

Imagine that the learner is confronted with the error in 43):

\[\text{input} \quad \text{winner} \sim \text{loser} \quad *\text{mid} \quad \text{Ident[\text{mid}]} \quad \text{Ident[\text{mid}]_\text{L}}\]

\[\text{bedat} \quad \text{bedat} \sim \text{bidat} \quad L \quad W \quad W\]

In order to resolve this error, the learner knows that *some* faithfulness must be installed above *mid; that is, they can choose between one of these two rankings:

---

20 See also the rankings in Strujke (2002) between other Faithfulness and her Existential Faith constraints.
44) Two ways to resolve the ERC in 43)

a) Ident(mid)-σ₁ >> *mid >> Ident(mid)
b) Ident(mid) >> *mid >> Ident(mid)-σ₁

Of these two rankings, 44a) is the more restrictive, because it accounts for the winner’s mid vowel while leaving the height of non-initial vowels up to markedness. This grammar is one in which mid vowels are allowed to surface faithfully only in initial syllables; elsewhere they are still ruled out by *mid:

45) The restrictive results of 44a)

a) initial mid vowels survive  b) ... but non-initial ones do not

<table>
<thead>
<tr>
<th>input</th>
<th>Ident(mid)-σ₁</th>
<th>*mid</th>
<th>Ident(mid)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>*bedat</td>
<td>*</td>
<td>*bedat</td>
</tr>
<tr>
<td>(ii)</td>
<td>bidat</td>
<td>!</td>
<td>*bidat</td>
</tr>
</tbody>
</table>

Assuming the other ranking in 44b), however, means that any input mid vowel will be able to surface faithfully:

46) The possible overgeneration of 44b):

a) initial mid vowels survive  b) ... and so do others!

<table>
<thead>
<tr>
<th>input</th>
<th>Ident(mid)-σ₁</th>
<th>*mid</th>
<th>Ident(mid)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>*bedat</td>
<td>*</td>
<td>*bedat</td>
</tr>
<tr>
<td>(ii)</td>
<td>bidat</td>
<td>!</td>
<td>*bidat</td>
</tr>
</tbody>
</table>

The problem with 46) is the usual superset problem: that if the target language does in fact only permit mid vowels in initial syllables, the grammar in 46) won’t cause any further mid vowel errors, and so won’t provide any evidence that an overly-

permisive language has been learned. If however the reverse error has been made – the learner has chosen the grammar of 45) instead of 46) – the learner will get evidence that they’ve made the wrong decision. Once they hear a mid vowel in a non-initial syllable, they’ll make an error that creates an ERC row like in 47) below. This error clearly demonstrates that installing Ident(mid)-σ₁ will not account for the target’s full range of marked vowels, so that Ident(mid) must be used:

47) an unambiguous ERC row that chooses the ranking in 46)

<table>
<thead>
<tr>
<th>input</th>
<th>Ident(mid)-σ₁</th>
<th>*mid</th>
<th>Ident(mid)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>*bedat</td>
<td>*</td>
<td>*bedat</td>
</tr>
<tr>
<td>(ii)</td>
<td>bidat</td>
<td>!</td>
<td>*bidat</td>
</tr>
</tbody>
</table>

The same relationship holds true of morphologically-specific faith, e.g.:

48) an ERC row adapted from Smith (1999)

<table>
<thead>
<tr>
<th>input</th>
<th>NoCoda</th>
<th>Max(Seg)-Rt</th>
<th>Max(Seg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>*bedat</td>
<td>*bedat</td>
<td>*beda</td>
</tr>
<tr>
<td>(ii)</td>
<td>*bedat</td>
<td>*beda</td>
<td>*beda</td>
</tr>
</tbody>
</table>

When presented with this error, the learner must choose the ranking in 49a) below, and not 49b). This ensures that if the target language only permits codas in roots, the learner will not mistakenly assume that affixes can have codas too.

49) The two ranking possibilities given 48)

a) Max(Seg)-Rt >> NoCoda >> Max(Seg) (restrictive)
b) Max(Seg) >> NoCoda >> Max(Seg)-Rt (not restrictive)
The conclusion, then, is that finding the most restrictive grammar compatible with a set of ERC rows depends in part on installing the most specific W-assigning faithfulness constraint above each L-assigning markedness constraint.

4.3 The problems of enforcing the Spec-F >> General-F bias

4.3.1 Language-specific relations between faithfulness constraints

As Prince and Tesar (2004) demonstrate, implementing a specific >> general faithfulness bias is not at all straightforward. The previous two biases were easy to enforce because they make reference to language-independent properties. Markedness or Faithfulness, Output-Output or Input-Output – these are definitional properties of a constraint. But there is not always something intrinsic to the definition of a constraint that puts it in a special to general relationship with another – and while many specific to general relations are universal, Prince and Tesar also point out that the stringency relations between faithfulness constraints can be contingent. In contingent cases, it is only the interaction of other high-ranking constraints that carve up the space of possible surface forms in such a way to make a particular faithfulness constraint less or more specific than another.

4.3.1.1 Prince and Tesar’s example

Here is the extent of the problem. Imagine that our learner has constructed the following ERC row:

<table>
<thead>
<tr>
<th>winner~loser</th>
<th>*mid</th>
<th>Ident-mid-σ1</th>
<th>Ident-mid-σ'</th>
</tr>
</thead>
<tbody>
<tr>
<td>kipa ~ kípa</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

Which of the IO-faithfulness constraints that prefer the winner should be installed? What follows is Prince and Tesar’s demonstration that given other facts about the target language (in particular, its pattern of stress assignment), these two faithfulness constraints might stand in either stringency relation.

In Language A, the correct generalization is that mid vowels only appear in initial syllables – that is, the true ranking in the language is in 51)

51) H_{LA}: \text{Id(mid)-σ}_1 >> *\text{mid} >> \text{Id(mid)-σ'}

Suppose further that this language assigns stress without fail to the initial syllable of every word; it also assign stresses to later syllables in longer words (this is the case in e.g. Pintupi; see Hayes, 1995 and references therein.) In such a grammar, the initial syllable context is in fact more specific than the stressed syllable context – because of how stress is assigned, every initial syllable is stressed, but not every stressed syllable is initial.

Now consider Language B, whose Support table also includes the entry in 50) but whose correct ranking is the reverse of Language A:

52) H_{LB}: \text{Id(mid)-σ'} >> *\text{mid} >> \text{Id(mid)-σ}_1

In language B, stress is always confined to the initial syllable of a word, but some words do not bear stress at all.\textsuperscript{21} As a result, these two contexts (and associated faithfulness

\textsuperscript{21}On the plausibility of such a language, Prince and Tesar (2004) cite the example of Seneca, (e.g. Michelson, 1988), where “stress behaves more like pitch accent [and] stressless words may occur”. Perhaps
In their search for the most restrictive grammar, the learners of Language A or Language B are in equally dangerous but opposite situations. In resolving this one error, the specificity relations between Ident(mid)-σ1 and Ident(mid)-σ' are crucial to choosing the subset grammar – but depending on the language either relation could hold. And the language-specific evidence as to which context is more specific than the other can’t be read off any faithfulness constraint to mid vowels, initial syllables or stressed syllables. Instead, these facts are only buried away in the constraint rankings that determine stress – Align-Head-L, Trochee vs. Iamb, and the like.

4.3.1.2 A morphological example

Given the centrality of this issue to my argument, it is worth seeing that this problem is not just a function of initial and stressed syllables. Contingent specificity relations between positional contexts will also emerge as the result of morphologically-specific constraints like Root-Faith. For this case, we can use the same ERC, only slightly modified:

53) a morphologically ambiguous ERC

<table>
<thead>
<tr>
<th>input</th>
<th>winner-loser</th>
<th>*[mid]</th>
<th>Ident-mid-σ1</th>
<th>Ident-mid-Root</th>
</tr>
</thead>
<tbody>
<tr>
<td>kepā</td>
<td>képā – kipa</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

The languages that could have created this error include the following two in which the stringency relation is crucial, but again in either direction. In Language A, there are no prefixes or pro-cliticizing elements, so every initial syllable is also a root syllable. But the reverse is not true, since roots can be bigger than a syllable. So in language A: initial syllables are a special case of root syllables.

In the second language, affairs are different. Language B again has no prefixes, but its roots are small – in fact, no longer than a syllable – meaning that root syllables are always initial syllables. Furthermore, this language has free-standing words that do not count as roots – i.e., function words can create their own Prosodic Words, so some initial syllables are not root syllables. Thus in language B, root syllables are a special case of initial syllables.

The resulting learning problem is just as in the previous section. Language A could restrict mid vowels only to initial syllables, and Language B could restrict mid vowels only to roots – both meaning that only the Ident constraint referring to the more specific context should be installed above *[mid]. But how can the learner know which Ident constraint that is? Again, the fact that root syllables are more or less specific than initial syllables is encoded only in the ranking of constraints that say nothing about the mid vowel that caused the error in 53), but rather in the ranking of constraints that e.g. align morphological roots and prosodic words (McCarthy and Prince, 1993 et seq.)

4.4 Interim Summary

In the face of the problem raised above, Prince and Tesar (2004) go so far as to suggest that positional faithfulness in fact be barred from CON, and replaced with positional markedness constraints instead (e.g. Zoll, 1998) whose specificity relations pose no problem for learning, as discussed in §3.3. Nevertheless, the claim of this
dissertation is that the kinds of evidence marshaled at the beginning of this discussion
(§4.1.1) require us to include positional faithfulness constraint in our typology, and
therefore that their consequences for restrictiveness must be accommodated by our
learner.22 And since the previous sections have demonstrated that a restrictive learner
must be biased to choose the faithfulness constraint possible with the most specific
context when installing W-preferring constraints: our learner will have to be able to
discover these relations.

Returning to the broader picture: sections 2 through 4 have presented the Biased
Constraint Demotion approach to restrictiveness, and the three ranking biases that I adopt
in my version of this algorithm. Before moving on, the next section returns to the role of
the Support in BCD learning, and uses the biases we’ve now seen to emphasize its key
role in the on-going quest for restrictiveness. Then, in section 6, I will return to the proper
treatment of the specific >> general faithfulness bias, given the problems just raised in
§4.3.

5. Returning to the role of the Support

As was emphasized at the beginning of this chapter, the Rich Base assumed in
Optimality Theory means that language-specific knowledge in an adult OT grammar is
instantiated fully in rankings, rather than in lexical items. Nevertheless, in the BCD
approach it is in fact the Support and not the constraint rankings that are the real locus of
learning over time. BCD is a function from Support data to a ranking; the BCD learner is
gradual and incremental at the level of the Support, but quick and flexible in its rankings.

22 See Beckman, Jessen and Ringen (2006) for an interesting different kind of argument for positional faith
over positional markedness.
The first point about such misparses is that, like other errors, they are all a function of the current ranking. An upcoming example is the syllabification of a word-medial, post-tonic cluster like [kâbla]. On the one hand, NoCoda prefers the complex onset parse, as in (51a). On the other hand, however, the constraint Stress-to-Weight (e.g. Hanson and Kiparsky 1996; Elenbaas, 1999; Elenbaas and Kager, 1999) prefers syllabifying this cluster as a coda-onset cluster, because it requires that stressed syllables be closed. What will decide between these syllabifications is thus relative re-ranking of these two constraints at the point when the learner hears this winner:

54) Choosing the syllabification of the winner

<table>
<thead>
<tr>
<th></th>
<th>[kâbla]</th>
<th>NoCoda</th>
<th>Stress-to-Weight</th>
<th>[kabla]</th>
<th>Stress-to-Weight</th>
<th>NoCoda</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>* kâbla</td>
<td></td>
<td>*</td>
<td>kâbla</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>* kabła</td>
<td></td>
<td>*</td>
<td>kabła</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

Note that since they are based on constraint rankings, winner misparses are not necessarily infrequent – that is, they are not one-time glitches. If the learner’s current grammar assigns the wrong syllable structure or other representation to a class of winners, they will continue to make these parsing errors until some further learning takes place.

The point here that is these winner misparses can prevent the correct acquisition of other aspects of the grammar, and that overcoming their influence requires remembering errors so one can undo the rankings that winner misparses caused. This is what the Support allows us to do.

5.2 How the Support allows BCD to overcome winner misparses

Imagine the learner is acquiring a language with coda devoicing, where onset obstruents can be voiced or voiceless but codas are always voiceless. This grammar requires the simple ranking in 48) below:

55) The target grammar

Ident[voice]-Onset >> *VoicedObs >> Ident[voice]

One thing the learner must do to learn this grammar is to correctly syllabify all voiced obstruents as onsets. If it does this correctly, then its ERC rows will look like 53) below:

56) Learning onset voicing: the right winner parse

winner – loser *VoicedObs Ident[voice]-Onset Ident[voice]

As we saw in 4.2, a bias for the most specific faithfulness constraint will ensure that this ERC row will teach the BCD learner the right ranking in 56). The problem illustrated in 54), however, is that an incorrect ranking of the constraints Stress-to-Weight and NoCoda could drive this learner to the coda-onset syllabification of this cluster. If the learner has adopted the 54b) ranking in which Stress-to-Weight chooses the coda-onset sequence, this will change the violations in their ERC row for kabla ~ kapla. In this grammar, the winner’s [b] and loser’s [p] of this cluster are syllabified as a coda, so Ident[voice]-Onset will not make a choice between them:

---

23 Thanks to John McCarthy for suggesting this example.
The only grammar that BCD can learn from this ERC row is the ranking below:

58) **The superset grammar learned from 57):**

\[\text{Ident[voice]} - \text{Ons} >> \text{Ident[voice]} >> \text{*VoicedObstruent}\]

And this ranking defines a superset language compared to 55), because it allows a spurious voicing contrast in coda position:

59) **The restrictiveness problem with 58):**

\[
\begin{array}{c|c|c|c}
\text{hypothesical} & \text{Ident[voice]} - \text{Ons} & \text{Ident[voice]} & \text{*VoicedObs} \\
\hline
\text{ə kā} & * & & \\
kāp & & & *
\end{array}
\]

So for the present, the learner has acquired a superset grammar – so long as they have the learning ERC row of 57) in their Support, every cycle of BCD re-ranking will generate a grammar with general Ident[voice] ranked too high.

But this error is not a permanent overgeneralization for the learner. When the learner gets evidence for the ranking NoCoda >> Stress-to-Weight, he or she will now have a way to update his or her Support entries – fixing the input and winner representations, and thereby calculating the correct constraint violations as in 56). (I do not go into the details here of how such further learning operates – see Tesar and Smolensky 2000’s notion of RIP and the acquisition of hidden structure like footing; see Tesar et al 2003 for their discussion of a proposal about later morphological reparsing of winners in the Support, using the process called Surgery.) But once it has happened, the first time they re-rank after that Support update, they will now choose the right ranking.24

Note that the learning cycle that gets the learner to the correct grammar will not come as a result of any voicing errors. Once BCD has chosen a grammar on the basis of the incorrect parse, it now allows voiced obstruents in all syllabic positions, so it cannot make any errors in phonotactic learning. The reason the BCD learner can nevertheless overcome these winner misparses, and revert to a subset grammar is that it stores the errors it is no longer making. When its ranking changes in a way that can affect old errors (e.g. re-syllabify them), it can re-calculate the relevant constraint violations and so use them to choose the newest, most restrictive grammar.

5.3 **A second example** 25

This example involves a different kind of winner misparse which has received considerable discussion: the footing of a trisyllabic word with medial stress, as either trochaic \(\sigma (\sigma \sigma)\) or \((\sigma \sigma \sigma)\) (see Tesar, 2000.)

Imagine our learner is now acquiring an iambic language with foot-initial strengthening: foot-initial stops must be aspirated, but all others are unaspirated. This language requires has the allophonic ranking in 60) below:

60) **The target grammar**

\[\text{Foot-Initial Aspiration >> *Aspirate >> Ident-[laryngeal]}\]

---

24 It is worth noting that choosing the right syllabification of a medial consonant sequence is far from easy. It may indeed depend on something quite subtle process that e.g. can be understood as triggered only by closed syllables, and which is not triggered in the first syllable of CVbV words with initial stress.

25 Thanks to Joe Pater for suggesting this example.
To correctly diagnose this allophonic ranking, the learner must again have the winner’s representations correct. If the learner has parsed all its feet into iambs, then its ERCs rows will look like 61), and BCD will be able to choose the right ranking.

61) Learning foot-initial aspiration: the right winner parse

<table>
<thead>
<tr>
<th>/pʰˈabóla/</th>
<th>*Aspirate</th>
<th>Ft-InitialAsp</th>
<th>Ident[laryng]</th>
</tr>
</thead>
</table>

Imagine however that the learner were to make the error above early on, at a point when foot form had not yet been decided. What will decide between these foot structures is the relative ranking of Trochee and Iamb:

62) Choosing the footing of the winner

(a)        (b)

<table>
<thead>
<tr>
<th>/pʰˈabóla/</th>
<th>Iamb</th>
<th>Trochee</th>
<th>/pʰˈabóla/</th>
<th>Trochee</th>
<th>Iamb</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pʰˈa(bóla)</td>
<td>*</td>
<td>*</td>
<td>pʰˈa(bóla)</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

If the learner has adopted the ranking in (b) instead of (a) (even just for this error), this will create a different ERC row, precisely with respect to the foot-initial aspiration constraint:

63) The misparsed ERC row

<table>
<thead>
<tr>
<th>/pʰˈabóla/</th>
<th>*Aspirate</th>
<th>Ft-InitialAsp</th>
<th>Ident[laryng]</th>
</tr>
</thead>
<tbody>
<tr>
<td>[pʰˈa(bóla)] – [pa(bóla)]</td>
<td>L</td>
<td>c</td>
<td>W</td>
</tr>
</tbody>
</table>

And the only grammar that BCD can learn from 63) is the superset one below:

64) The superset grammar:

Ft-InitialAsp > Ident[laryngeal] >> *Aspirate

This ranking defines a superset language compared to the target in 60), because it allows a spurious laryngeal contrast outside the foot-initial position:

65) The restrictiveness problem:

<table>
<thead>
<tr>
<th>hypothetical</th>
<th>Ft-InitialAsp</th>
<th>Ident[laryng]</th>
<th>*Aspirate</th>
</tr>
</thead>
<tbody>
<tr>
<td>∅ (bopʰa)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>⊕ (bopa)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Again: this winner misparse has led our learner to acquire a superset grammar. But once Trochee has been properly re-ranked above Iamb, the learner can re-evaluate their ERC rows like 61) to look instead like 63), which will finally lead to the right allophonic ranking.

5.4 Summary

This section has highlighted a crucial role of stored errors in biased learning – the escape from superset grammars when early learning data is re-interpreted. I will also return to this point in later chapters, when I demonstrate the trouble that a memory-less learner like the GLA has in keeping restrictive despite winner misparses (chapter 3 §4; chapter 4 §7.3.2.)

With the centrality of the Support fully in mind, we can now return to the problem of the specific-F ranking bias, and see how learners can use their current Support to handle even contingent F-subset relations.
6. The proposal: finding the most specific IO-Faith constraint

Section 4 provided arguments (i) that our theory should contain positional IO-faithfulness constraints, (ii) that their presence in CON requires a bias for BCD to rank the most specific constraints as high as possible, and (iii) that specific-to-general relations between faithfulness constraints are not all universal, and can differ crucially from language to language. With all these claims in mind, this section presents the proposal for properly constructing this bias.

6.1 The goal: determining subset relations between the contexts of faith

To adequately determine specific-to-general faithfulness relations, the learner must somehow access information about the rest of the language being learned, which our BCD learner is storing in the Support. Central to what follows is the idea that to discover the specificity of faithfulness constraints, learners must abstract away from particular constraints and instead consider the contexts of those constraints. This move is somewhat subtle, but it represents a rather different approach to the problem than the one envisioned in Prince and Tesar (2004)’s discussion, so I will endeavour to clarify the matter below.

First, the problem. If a language has a crucial ranking of a specific faithfulness constraint above a more general one, it is because something marked only appears in the specific position but it banned in the general one. What this means for the learner is that in errors during phonotactic learning, the constraint violations assigned by the specific and general IO-Faith constraints will be identical: when either assigns a W, they both will:

<table>
<thead>
<tr>
<th>input</th>
<th>winner – loser</th>
<th>*mid</th>
<th>Ident[mid]-Seg</th>
<th>Ident[mid]-σ1</th>
</tr>
</thead>
<tbody>
<tr>
<td>bedat/</td>
<td>bedat – bidat</td>
<td>W</td>
<td>L</td>
<td>W</td>
</tr>
</tbody>
</table>

No errors will exist in which Ident(mid) assigns a W but Ident(mid)-σ1 does not. Thus, our learner is not going to learn that Ident(mid)-σ1 is more specific than Ident[mid] by examining this ERC row – something must be abstracted away from.

In my proposal, the learner’s attention is directed away from the Ws assigned by the faithfulness constraints in 66), and even away from mid vowels, and instead aimed at the contexts “Seg” and “σ1”. The learner will examine the observed winners of their language, and determine whether these positional contexts sit in a subset/superset relation or not; from that information they will choose constraints to install.

6.1.1 Constraint stringency vs. context specificity

The way of calculating the specific-to-general bias that I will propose relies on context specificity, rather than constraint stringency itself, to guide the learner’s ranking decisions.\(^{26}\) It is important to see that the specificity of contexts does not translate straight to the stringency of constraints – because faithfulness constraints have both contexts and also banned mappings. In 66) above these two properties do line up, because the two faithfulness constraints at hand are both relative to the vowel feature [mid]. Thus we can say both that the context ‘σ1’ is more specific than “Seg”, and also that the Ident[mid]-σ1 constraint is less stringent than Ident[mid]-Seg.

\(^{26}\) The method Hayes (2004) uses to impose his Favour Specificity bias also uses context specificity rather than constraint stringency; see section 7.2 for more details.
But stringency relations do not hold between constraints that protect different features. For example, Ident[mid]-σ1 is not less stringent than Ident[voice]-Seg because they penalize completely different mappings, so each can be violated independent of the other:

<table>
<thead>
<tr>
<th>/peg/</th>
<th>Ident[mid]-σ1</th>
<th>Ident[voice]-Seg</th>
</tr>
</thead>
<tbody>
<tr>
<td>peg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pig</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>pek</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>pik</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

I will argue in section 6.5 below that there is good reason to focus on contexts rather than constraints – because even when abstracting away from the unenlightening cases like 66), constraint violations can be misleading when it comes to contingent stringency relationships. To understand the proposal, however, it is enough to remember that this search for subset/superset relations will be focused on contexts, and only then will apply the search’s findings to faithfulness constraints and their ranking.

### 6.1.2 Outline of the proposal

To summarize so far: what we need is a way to detect any specificity relationship between any two faithfulness contexts C1 and C2. Broadly speaking, this search is going to involve looking at each instance of C1 among the language’s winners and seeing whether it is also an instance of C2, and then vice versa. If the two contexts do not stand in a specific-to-general relation, we will only need to come across the two relevant pieces of evidence to determine no such relation exists. To determine that there is such a relation, however, our search will have to continue until all winners have been examined – to be sure that every known instance of C1 is also a case of C2.

As we saw in section 4, central to Prince and Tesar’s skepticism about the specific faithfulness bias is the existence of contingent faithfulness stringency – in the present terms, contexts can sit in contingent specificity relations. However, many contexts always stand in specificity relation, and it could well be argued that searching the entire Support for evidence of such relations is rather inefficient. One universal relation is simply that the most general faithfulness context – what I have been calling simply “Seg” – is less specific that any faithfulness constraint above the segmental level, which references either a prosodic or morphological category:

| Id(mid)-Onset, Id(mid)-σ₁ … | Id(round)-Rt, Id(round)-Noun … |
| Id(mid)-segment | Id(round)-Segment |

The same point can be made at the featural level – that the context “Seg” is more general than any more specific combination of features (e.g. Ident(voice)-Labial, Ident(voice)-Labial and Dorsal.) In fact a stronger claim can be made at the featural level, by adopting the fairly standard OT assumption that there are no language-specific meanings to featural combinations – i.e. that the context “Labial and Dorsal” refers to the same set of segments in every language.27 This means that the relationships between featural contexts is definitional and language-independent, and so all their subset and

---

27 For the alternative type of view, see e.g. Rice and Avery 1989, 1991.
superset relations can merely be read off their constraint definitions. This is illustrated in (69) below:

(69) **Two examples of universal featural context specificity**

<table>
<thead>
<tr>
<th>Context 1</th>
<th>Context 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IdentPlace-(Stop or Fricative)</td>
<td>IdentVoice-(Nasal)</td>
</tr>
<tr>
<td><strong>is less stringent than</strong></td>
<td><strong>is less stringent than</strong></td>
</tr>
<tr>
<td>IdentPlace-(Stop or Fricative or Nasal)</td>
<td>IdentVoice-(Nasal or Oral Stop)</td>
</tr>
</tbody>
</table>

As a result of these universals, my approach to calculating IO-faith specificity has two steps. When considering a set of faithfulness constraints, the learner will first ‘pre-compile’ the specificity between their contexts using universal properties of CON – defined below as the kind and number of ‘contextual arguments’ each takes (explanation to follow.) If this first pass underdetermines the specific-to-general relations (as it will in the case of e.g. Ident(F)-σ° and Ident(F)-σ₁) – the learner will then turn to the Support’s winners, and evaluate contexts one by one. This second step will require a tool over which the learner can calculate these relationships, which I will call here the Context Table. To summarize:

(70) **The method for detecting IO-faith Specificity**

- (a) Is there a universal relation between contexts X and Y? (the Context Arguments test)
- (b) Does the Support reveal a contingent relation between contexts X and Y? (the Contingent Specificity test)

In the sections that I follow I demonstrate how and why these two steps work; integrating them into a BCD-style ranking bias will be the province of section 7.

Before continuing, a remark about efficiency and computation may be in order. Given that the first step of this method establishes contextual relations that are universal, it might be something of a pedantic pity to re-calculate whether e.g. segments that are onsets are a special case of segments every time the learner needs to choose the ranking Ident(F)-Onset >> M >> Ident(F)-Seg. As an alternative, one could either hardwire the learner with the knowledge of these context subset relations, or else ask the learner to apply step 1 every time they need to but then store the results to be used in all subsequent rankings.

If these two alternative approaches were to be implemented computationally, it seems likely that a constant recalculation of universal context specificity would add unnecessary effort to the learner’s task. The only reason to do so would seem to be a purely formal interest in hard-wiring as little into the learner as possible – beyond this aesthetic concern, I leave the matter open.²⁹

### 6.2 The first step: finding universal specificity relations

The schematics in 68) and 69) showed the two kinds of universal specific-to-general relations we want our learner to notice just from constraint definitions, or rather the definition of their contexts.

²⁹ One substantive issue with hardwiring these universal specificity relations might arise when comparing constraints with multiple contexts, some but not all of which are in universal stringency relations, and thereby deciding e.g. whether Ident(mid)-Root-Seg and Ident(mid)-σ₁-Labial are in a stringency relationship. Whether the relevant constraints are ever in conflict in a ranking situation is an empirical question to which I do not know the answer.
First, we must decide what a context is. All work on positional faithfulness is concerned with giving constraints the right structural context – input initial syllable, output onset, and so forth – as well as its banned unfaithful mapping (voicing mismatch, missing correspondent, etc.) In this work, the contextual arguments of constraints are always at the level of the segment, or above: either a prosodic or morphological category. Some representative examples of constraints and their context arguments are given below:

71) constraint positional context argument(s)
(a) MAX segment
(b) DEP-{V} vowel & segment
(c) IDENT[voice] segment
(d) IO-IDEN[t][lab]-Ons onset & segment
(e) IO-IDEN[t][phar]-Rt root & segment
(f) IO-IDEN[t][mid]-Rt-σ initial syllable & root & segment

As 71) shows, all the positional constraints have been defined with the argument segment as well as something else. This is important for one thing because it allows the comparison of phonological and morphological contexts, whose affiliations only overlap at the segmental level. With the arguments defined this way, we can find the most specific members of a set of positional contexts merely by finding proper subsets:

72) the Positional Context Arguments test
GIVEN: two positional context arguments, P1 and P2
IF: if P1 is a proper subset of P2 (that is: every member of P1 is also a member of P2, but some member of P2 is not a member of P1)
THEN: P1 is less specific than P2

By the test in 72), the constraints in 71) include three specificity relations, that are clearly the right ones:

(i) MAX (a)’s context is less specific than those of constraints (b),(d)-(f)
(ii) IDENT (c)’s context is less specific than those of constraints (b),(d)-(f)
(iii) IO-Ident-Rt (e)’s context is less specific than that of IO-Ident-Rt-σ

I am also assuming that faithfulness constraints can be relativized to featural contexts. Thus, we will need to determine specificity relations among the subsegmental context arguments of constraints. One current view of featural faithfulness comes from the theory of de Lacy (2002), in which faithfulness constraints protect features in direct proportion to their markedness, and where featural markedness directly reflects stringent markedness scales. As an example, the featural markedness scale on major places of articulation in 70) below results in a set of IDENT[place] constraints, which increase in stringency from the most specific to the most general:

74) Place of Articulation Markedness Scale, from most to least marked:
(dorsal > labial > coronal > glottal)

Ph.D. dissertation, UMass Amherst
75) constraint examples  subsegmental context argument(s)
(a) IDENT\[place\]-dors   dorsal
(b) IDENT\[place\]-dors, lab   dorsal or labial
(c) IDENT\[place\]-dors, lab, cor   dorsal or labial, or coronal
(d) IDENT\[place\]     dorsal or labial or coronal or glottal

This theory's faithfulness constraints wear their stringency relations on their sleeves. But notice that the relationship between the number of arguments and the specificity of a constraint is opposite to the prosodic domain: here, each argument adds another featural context for faithfulness to apply to, rather than further restricting its application. As such, the test for subsegmental arguments equates decreased specificity with supersets rather than subsets:

76) the Subsegmental Context Arguments test
GIVEN: two subsegmental context arguments, S1 and S2
IF:    if S1 is a proper superset of S2
THEN:  S1 is less specific than S2

By this reckoning, we can calculate that 75(d)'s context is less specific than all three other constraints; that 75(c)'s context is less specific than a) and b)'s, and so on.

6.3 The second step: finding contingent specificity relations

We now have seen the straightforward way of determining the specific-to-general relations inherent to constraint definitions. In this section I turn to the method of determining contingent specific-to-general relations, using a tool I will call the Context Table (CT). Note that the method for building such tables, though to a slightly different end, was also proposed by Hayes (2004), footnote 31, and that my CTs were (re)designed with his proposal in mind.

77) A Context Table, not yet filled in

<table>
<thead>
<tr>
<th></th>
<th>σ'</th>
<th>onset</th>
</tr>
</thead>
<tbody>
<tr>
<td>σ'</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>onset</td>
<td>?</td>
<td></td>
</tr>
</tbody>
</table>

At the left of each row and the top of each column are a series of contexts. In a completed CT, the two cells marked "?" will either be filled in with a check or left blank. To complete the table, the learner must find every instance of each context in every winner. For illustration purposes, imagine the learner has added only three ERC rows to the Support thus far, and that the resulting winners are as in 78):

78) A set of stored words:
   (a) pát  (b) ibák  (c) páda

Here is how a context table is completed. To fill in each cell, the learner asks whether each instance of phonological material in the context of row x is also in the context of column y:

79) The Context Table procedure
GIVEN: For a set of winners, and a context table with rows r1 to rν, columns c1 to cμ and cells <row, column>
IF:    some segment is in the context r but not in the context of c1
THEN:  put a mark in cell <i,j>

A Context Table is a chart that keeps track of whether a faithfulness context can occur independent of another, in a given set of words:
In 77), there are two contexts, so the ‘if’ statement of 79) above ranges over two questions: “is every segment in a stressed syllable also in an onset?” and “is every segment in an onset also in a stressed syllable?” Given the first question: the first segment in a stressed syllable that is not an onset (say, the á of [pat] in 78a) will cause the learner to update this context table as in 80):

80)  

<table>
<thead>
<tr>
<th>σ’</th>
<th>onset</th>
<th>σ’ onset</th>
<th>D onset</th>
</tr>
</thead>
<tbody>
<tr>
<td>onset</td>
<td>?</td>
<td>✔</td>
<td></td>
</tr>
</tbody>
</table>

To answer the second question: once the learner comes across the onset of an unstressed syllable (like the [d] of 78c), they will add a second mark to their table:

81)  

<table>
<thead>
<tr>
<th>σ’</th>
<th>onset</th>
<th>σ’ onset</th>
<th>D onset</th>
<th>D onset</th>
</tr>
</thead>
<tbody>
<tr>
<td>onset</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Reading context tables to answer the specificity question works as follows:

82)  

The Contingent Specificity test

GIVEN: two context arguments, C1 and C2, and a context table with cells <row, column>

IF: cell <C1,C2> of a Context Table has a mark while cell <C2,C1> does not

THEN: C1 is less specific than C2

As soon as the learner has put marks in cells <C1,C2> and <C2,C1> of a table, the test above already tells them that no stringency relation holds between two constraints – the search is over. However, if at the end of the search these two cells are asymmetric (one has a mark, and the other doesn’t) the learner will have found a contingent specificity relation.

To illustrate this, imagine we provide our learner with a novel language, which has the following two attested properties. First: as in Pintupi discussed in §4.3.1.1, stress is assigned using trochaic feet built L-to-R, meaning that initial syllables are always stressed, as well as every odd syllable after that. This means that initial syllables are a special case of stressed syllables. Second, mid vowels in this language are restricted to initial syllables only – this is the case in Shona: see esp. Beckman (1998) and references cited therein. So let’s call this language Shontupi. With respect to the distribution of mid vowels, the necessary Shontupi ranking is:

83)  

Ident[Mid], Ident[Mid]σ  >>  *Mid >> Ident[Mid], Ident[Mid], etc.

And in the course of acquiring Shontupi, our learner encounters the by-now familiar ERC row repeated in 84):

84)  

```
/képa/ *[mid] Ident [mid]σ  
/képa ~ kípa  L W W
```

Based on this error, our learner can build a context table. To do so, we must have some winners to examine, so:

85)  

A representative set of Shontupi words

- képa
- tipu
- kóphilà
- tabukida
In examining these words using the reasoning from the context table procedure above, the learner will discover e.g. that the segments [la] of ‘kópilá’ are in the context [σ’] but not [σ₁]. This means that it will enter a mark in cell <σ’, σ₁>:

86) The resulting context table

 On the other hand, it will never come across a word in which some segment is in an initial syllable, but not a stressed syllable. Thus the table in 86) will be its final context table, to which it will apply the Contingent Stringency test:

87) The Contingent Stringency test, applied to 86)

SINCE: cell <σ’, σ₁> cell has a mark while cell <σ₁, σ’> does not
THEN: context [σ’] is less specific than the context [σ₁]

And with this knowledge, the specific >> general faith bias our learner will be armed with in section 7 will correctly resolve errors like the one in 84) by installing just the more specific constraint, Ident[mid]-σ₁.

6.4 Why context tables are dynamic

The context tables that I have proposed here are dynamic. They are built on-the-fly – constructed as biases demands, used once to choose between a set of Faithfulness constraints at one particular stratum and then forgotten, and are not stored in any learning memory. In other words, context tables are NOT like the Support, in that they are not the learner’s gradual lexicon of contextual asymmetries encountered so far, built up incrementally over time. This section demonstrates the reason to build CTs dynamically rather than incrementally: to prevent redundancy in the recovery from winner misparses that will add extra marks to early context tables.

6.4.1 What can go wrong in a context table?

Two things can go wrong in the building of a context table. Compared to a hypothetical correct table, either a missing mark can be absent from a cell where one should be (89a), or an extra mark can be present in a cell where one shouldn’t be (89b):

88) The correct Context Table for a language

89) a) A missing mark  b) An extra mark

When will these mistakes get made, and what kinds of restrictiveness problems can they cause?

First: errors of missing marks are easy to make, but they don’t cause any permanent restrictiveness problems on either the dynamic or incremental approaches. For example, the learner of Shontupi who hasn’t seen any words longer than two syllables yet will only have seen words with one initial and one stressed syllable – being the exact same syllable in each word. If at this point they build a initial syllable/stressed syllable
context table, they will build the one in 89a), missing a mark, and so they will have no stringency reason not to install e.g. Ident-[mid] σ', which can unfortunately build them a superset grammar.

However, this missing mark will appear as soon the learner encounters a word like e.g. képilà, in which the learner can see that stressed syllables need not be initial. And regardless of whether the context table that encodes this discovery is being built from scratch when required by the BCD algorithm, or being augmented incrementally, the next time the learner uses the Context Table procedure it will correctly include the mark and so require the correct installation of only Ident-[mid]-σ.

Unlike missing marks, extra marks will get into a context table not by having insufficient data but through the structural misanalyses that I have called ‘winner misparses’. In the present case, a relevant winner misparse would make the learner of Shontupi incorrectly believe they’d heard an unstressed initial syllable. This could result from misparsing a two-word sequence (clitic-noun, preposition-verb etc.) as one word, making it appear that stress falls on the second syllable of this false “word”:

90) A winner misparse
   a) correct parse:   [ba] [képilà]
   b) potential misparse:  [båképilà]

This misparse will result in a CT with an extra mark in the top right-hand corner:

91) the extra mark caused by a winner misparse

| σ' | σ' |
|HM|HM|
| σ|σ| (ké)
|σ'|σ'| (ké) |

Like with the missing mark, this CT wrongly assures the learner that there is no stringency relationship between initial and stressed syllable contexts, and so creates the possibility that Ident-σ' constraints will be installed when Ident-σ₁ constraints should be used.

As I argued in previous sections, winner misparses are a likely part of the learning process, because getting the right structural analyses of words depends on grammatical and lexical properties that are not available anywhere in the acoustic signal. Another such example where a winner misparse would choose a superset grammar comes from a language in which stressed syllables are always in the root. If the learner of this language misparses a word’s root-affix boundary as in 92b) below, the apparent fact that stress can appear on an affix would cause the creation of the incorrect context table in 93b):

92) A morphological misparse
   a) correct parse:   [σ' a a [σ root] word
   b) potential misparse:  [σ affix [σ σ root] word

93) a) the correct CT
   b) the extra mark CT

(The other mark, which registers the fact that root syllables need not also be stressed syllables, will be correctly drawn in response to either parse – because in both winners the second and third syllables are in the root but unstressed.)
6.4.2 Overcoming extra marks in a context table

Under the dynamic approach to context tables, extra marks are just hiccoughs in data processing – in the same way that winner misparses are themselves. If the Shontupi learner draws a misleading CT because of a bad parse, it may cause the temporary construction of a superset grammar, but this error will only last as long as the bad parse does. Once a word is reparsed and its Support entry has been corrected (e.g. 93b has been replaced with 93a), the next cycle of learning will require a new CT to be drawn and the right stringency relations will emerge.

In an incremental approach, however, extra marks would be somewhat messier to overcome, because they’d require a separate clean-up strategy in the CT-building mechanism. Once the learner had realized that [ba képilà] was in fact two words, she would have to entertain the re-calculation of every marked cell one of whose structural contexts was in the re-parsed word.32

Furthermore: making context tables dynamic leaves the Support as the one true and constant repository of learning data, in keeping with (at least part of) the BCD spirit. As the Support grows and changes, its emerging knowledge will influence both the frequent re-ranking of constraints and the frequent recalculation of dynamic context tables, both when prompted by the learning algorithm.

6.5 Why contingent specificity cannot be learned from Ls and Ws

In this section I demonstrate why contingent specificity relations cannot come from the comparison of the ERC rows themselves – that is, why they cannot be assessed by comparing the behaviour of faithfulness constraints rather than contexts (recall the discussion in section 6.1 above on the difference.)

What the learner wants to know about the target language is whether faithfulness in one of the two contexts under consideration implies faithfulness in the other, but not vice versa. So with the contexts of initial and stressed syllables in mind, one approach might have been to look across the Support for winner-loser pairs in which faithfulness to some feature is decisive in one of the two domains (assigning a W), but ambivalent in the other (assigning an e).

To give a concrete example that can demonstrate this approach’s failure, let us continue to imagine that our target language is Pintupi in which initial syllables are a special case of stressed syllables. So in looking for asymmetric ERC rows as suggested in the previous paragraph, the learner will be hoping for examples like 94) below:

94) An asymmetric ERC row

<table>
<thead>
<tr>
<th></th>
<th>front+rd</th>
<th>Ident-hi-σ1</th>
<th>Ident-hi-σ'</th>
</tr>
</thead>
<tbody>
<tr>
<td>kipy</td>
<td>L</td>
<td>e</td>
<td>W</td>
</tr>
</tbody>
</table>

In this approach, this ERC would act as a hint that Ident(hi)-σ1 is a more stringent constraint in the target language than Ident(hi)-σ'. This in turn would translate into a specificity relation between the two contexts [ŋ] and [œ'].

As it turns out, however, the nature of positional faithfulness makes this kind of evidence from ERC rows unreliable – even contradictory. Because when searching the
Support for asymmetric rows like 94), an unlucky learner of Pintupi might make the error in 95a), and build the ERC in 95b) as a result:

95a)  An unfortunate error:

<table>
<thead>
<tr>
<th></th>
<th>*front+rd</th>
<th>Ident(hi)−σ1</th>
<th>Ident(hi)−σ'</th>
</tr>
</thead>
<tbody>
<tr>
<td>kýpa</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>−kipá</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

95b)  The resulting perverse ERC

<table>
<thead>
<tr>
<th></th>
<th>*front+rd</th>
<th>Ident(hi)−σ1</th>
<th>Ident(hi)−σ'</th>
</tr>
</thead>
<tbody>
<tr>
<td>kýpa</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kipá</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In this language, this ERC demonstrates the reverse asymmetry compared to 94). As such, it is completely misleading to the learner: it suggests that stressed syllables are a special case of initial syllables! As we know that the target language’s initial syllables are always stressed, this can’t be right.

The crucial problem with using an ERC row like 95b) to reason in the way we did previously with 94) comes from the differences between their winners and losers. In 95b), the loser differs from the winner in its violation of Ident(hi)−σ1, because it has unrounded the initial syllable. However, this loser also differs from the winner by shifting stress onto the second syllable. And it is because of this second change that the loser does not violate Ident(hi)−σ': not because the input’s stressed syllable height has been preserved, but because the input’s stress has been moved. This problem is in some sense inherent to a pathology in the definition of positional faithfulness – one which has been raised by a number of authors33 and which remains unresolved.

---

33 See Beckman (1998) citing Rolf Noyer, as well as Wilson (2000).

7.  Implementing the Spec-F >> Gen-F bias

7.1  A working BCD algorithm

With all the results of the chapter so far, we are now in a position to put together our rankings biases into a working BCD algorithm. To understand how this works, we need one more piece of Prince and Tesar’s proposal; I will describe it here rather briefly, but the reader who is unfamiliar with BCD is encouraged to consult the much more thorough explication of these ideas in Prince and Tesar’s work.

We have already seen that ranking biases drive the learner to install e.g. markedness higher than faithfulness when possible. To get from ranking biases to a BCD algorithm, we must see how the learner stays as close to their biases as possible even when the data makes installing any of the preferred constraints impossible.

The leading idea from Prince and Tesar on this aspect of the algorithm is that the learner should install just as many of the dispreferred constraints in the current stratum as will allow the installation of preferred constraints in the next stratum. To take the markedness >> IO-faith bias: if there is more than one set of minimal IO faithfulness constraints that can be installed in stratum n that will allow some markedness constraints to be installed in stratum n+1, then the learner chooses the set that allows the most markedness constraints to be installed. In their BCD version, this idea is enforced only with respect to markedness >> IO-faith, but it can be generalized here to help enforce the OO-faith >> markedness bias as well.34

With this final piece, we can now see the prose version of the BCD algorithm I will be assuming for the rest of the dissertation. It is given in 96) below: while this

---

34 See Prince and Tesar 2004’s definitions of ‘Smallest Effective F-Sets and Richest Markedness Cascades for the details.
version is given in my own wording. I wish to be explicit that the BCD method itself is
in no way innovated beyond that of Prince and Tesar (2004) – my additions are just to
add the OO-faith bias as step 1, and to include the specific >> general IO-faith bias in the
way that I have.

96) My BCD algorithm

Step 1: Install all OO-Faith constraints that prefer no Ls
   a) if any can be installed, move onto step 3
   b) if none are left to installed, move onto step 2
   c) if some are left but none prefers no Ls,
      i) Install the smallest set of W-prefering Markedness constraints that will
         allow the installation of the most OO-faith constraints in the next
         consecutive strata35, and move onto step 3

Step 2: Install all Markedness constraints that prefer no Ls
   a) if any can be installed, move onto step 3
   b) if none are left to installed, move onto step 4
   c) if some are left but none prefers no Ls,
      i) Find the set of context arguments of all W-prefering IO-Faith
         constraints. If there is only W-preferer, install it and move onto step 3,
         otherwise
      ii) Determine all the specific-to-general relations among their contexts,
          using the Context Arguments and Contingent Specificity tests, and find
          the resulting set of W-prefering IO-faith constraints with the most specific
          context arguments, and then
      iii) Install the smallest set of these W-prefering IO-faith constraints that
          will allow the installation of the most Markedness constraints in the next
          consecutive strata and move onto step 3

35 Again, see the definition of ‘Richest Markedness Cascade’ in Prince and Tesar (2004) to understand what
   “the next consecutive strata” means.
The concern is what faithfulness constraint will be used to protect the fricative’s voicing in [az.ba].

In the illustration that Prince and Tesar originally provided and which I will follow, there are three Markedness constraints: two that penalize voiced obstruents (*b and *z), and one that requires voicing agreement between obstruent clusters (Agree(voice)). There are also four faithfulness constraints: defined just as Ident(b), Ident(z), and their onset-only versions, which we can construe as constraints protecting voicing. With these constraints and this lexicon the learner’s errors will all result from devoicing, in the three contexts that voicing appears: onset stops, coda stops, and coda fricatives. Thus, the Azba learner’s set of ERC rows looks like this:

98) The Azba learner’s Support

<table>
<thead>
<tr>
<th></th>
<th>Agree (voice)</th>
<th>*b</th>
<th>*z</th>
<th>Ident(b) -Onset</th>
<th>Ident(z) -Onset</th>
<th>Ident(b)</th>
<th>Ident(z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i)</td>
<td>ab ~ ap</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td>W</td>
</tr>
<tr>
<td>(ii)</td>
<td>ba ~ pa</td>
<td>e</td>
<td>L</td>
<td>e</td>
<td>e</td>
<td>W</td>
<td>e</td>
</tr>
<tr>
<td>(iii)</td>
<td>azba ~ aspa</td>
<td>e</td>
<td>L</td>
<td>L</td>
<td>W</td>
<td>e</td>
<td>W</td>
</tr>
</tbody>
</table>

From this Support set, the BCD learner will first install Agree(voice) in the top stratum. But since Agree(voice) assigns no Ws it resolves no errors, and both remaining markedness constraints *b and *z both assign Ls. So what IO-faithfulness constraint(s) should the learner choose?

7.2.1 Using a context-based F-specificity bias

In the BCD algorithm that I defined in 96), the learner will indeed choose the right constraint. This is partly because my learner computes specific to general relations via tests across contexts, not constraints. It is not necessary for a more specific version of a particular constraint (like Ident(z)-onset) to assign Ws in order to notice that Ident(z) has a very general context. This is illustrated below: after having assigned Agree(voice) to its first stratum, my learner will use its third step to install a constraint in the second stratum and come up with the right choice:

99) BCD learning of Azba with a specific-F contexts bias

(Since there are no OO-faith constraints under discussion here, I skip Step 1)

To build stratum 1:

Step 2: Install all Markedness constraints that prefer no Ls

Resulting stratum:

Step 3: Errors removed from the Support: none.

Error remaining: ba ~ pa, ab ~ ap, azba ~ aspa

To build stratum 2:

Step 2: Install all Markedness constraints that prefer no Ls

c) since some are left but none prefers no Ls:
i) Find the set of context arguments of all W-prefering IO-Faith constraints.
   W-prefering constraints: \{Ident(b)-Onset, Ident(b), Ident(z)\}
   Context arguments: \{Onset, Seg\}

ii) Determine all the specific-to-general relations among their contexts, using the Context Arguments and Contingent Specificity tests, and find the resulting set of W-prefering IO-faith constraints with the most specific context arguments.
   Resulting relations: Seg is less specific than Onset

iii) Install the smallest set of these W-prefering IO-faith constraints that allows the installation of the most Markedness constraints in the next consecutive stratum and move onto step 3
   Resulting stratum: Ident(b)-Onset

Step 3: Errors removed from the Support: ba~pa, azba ~ aspa
Error remaining: ab ~ ap

Now everything is smooth sailing. The only remaining error is 98(i), whose ERC row contains only one W-prefering IO-faith constraint, Ident(b). And after installing Ident(b) in the third stratum (by Step 3i), the learner will have resolved all the errors in the Support, so they are free to install all remaining markedness constraints in the fourth stratum (*b) and (*z), and then all remaining IO-faith constraints in the final stratum (Ident(z) and Ident(z)-Onset).

I spell out these remaining steps of the algorithm below just for completeness. But the crucial point is that this learner has installed all faithfulness constraints related to voiced fricatives (z) below the markedness constraint *z, and so chosen the most restrictive grammar consistent with the data:

100) BCD learning of Azba with a specific-F contexts bias, part two
    (continuing to skip Step 1, as in part one)

To build stratum 3:

Step 2: Install all Markedness constraints that prefer no Ls
   c) since some are left but none prefers no Ls:
      i) Find the set of context arguments of all W-prefering IO-Faith constraints. If there is only one W-preferer install it.
      Resulting stratum: Ident(b)

Step 3: Errors removed from the Support: ab ~ ap
Error remaining: none.

To build stratum 4:

Step 2: Install all Markedness constraints that prefer no Ls
   Resulting stratum: *b, *z

To build stratum 5:

Step 2: Install all Markedness constraints that prefer no Ls
   b) since none are left to installed, move onto step 4:

Step 4: Install all remaining IO-faith constraints in the bottom stratum, and END
   Resulting stratum: Ident(z)-Onset, Ident(z)

101) The final Azba ranking:
    Agree(voice) >> Ident(b)-Onset >> Ident(b) >> *b, *z >> Ident(z)-Onset, Ident(z)

7.2.2 Using a constraint-based F-stringency bias: Hayes’ simulation

Appendix A of Hayes (2004) discusses the Azba example, from both the perspective of his own Low-Faithfulness Constraint Demotion algorithm as well as a version of BCD that includes a version of the specific >> general faithfulness bias. He concludes there that if the BCD is equipped with this bias, the Azba-learning child will indeed correctly choose to install Ident(b)-Onset. His demonstration of this, using both algorithms, appears on his website.37

37 http://www.linguistics.ucla.edu/people/hayes/Acquisition/AzbaSpecificityBCD.htm
Hayes’ version of the specific faithfulness bias is like the one I have proposed in that he determines the specificity of contexts by looking at winners, but the way the bias uses that specificity information is somewhat different. Once his learner determines that no more markedness constraints can be installed, it determines the stringency relations of the faithfulness constraints themselves – and not just of all W-prefering constraints, but of all unranked IO-faithfulness constraints. It then and rules out all the more general ones, before considering which of the remaining constraints are W-prefering.

In the Azba case, this means that both the general Ident constraints are ruled out because their Ident-Onset constraints are less stringent. Then, choosing a W-prefering faithfulness constraint necessarily picks the Ident(b)-Onset constraint, because Ident(z)-Onset does not prefer any winners (it only assigns es.)

To see that this works, I apply this approach below to just build the first two strata of the Azba grammar:

<table>
<thead>
<tr>
<th></th>
<th>Agree (voice)</th>
<th>*b</th>
<th>*z</th>
<th>Ident(b)-Onset</th>
<th>Ident(z)-Onset</th>
<th>Ident(b)</th>
<th>Ident(z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) ab ~ ap</td>
<td>e</td>
<td>L</td>
<td>e</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td></td>
</tr>
<tr>
<td>(ii) ba ~ pa</td>
<td>e</td>
<td>L</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td>e</td>
<td></td>
</tr>
<tr>
<td>(iii) azba ~ aspa</td>
<td>e</td>
<td>L</td>
<td>L</td>
<td>W</td>
<td>e</td>
<td>W</td>
<td></td>
</tr>
</tbody>
</table>

To build stratum 1

Step 1: Install all Markedness constraints that prefer no Ls

Resulting stratum 1: Agree(voice)

To build stratum 2

Step 1: Install all Markedness constraints that prefer no Ls

Resulting stratum 2: -- empty --

102) BCD learning of Azba with an all-F stringent constraint bias (adapted from: http://www.linguistics.ucla.edu/people/hayes/Acquisition/AzbaSpecificityBCD.htm)

7.2.3 Prince and Tesar (2004) on the Azba language

A third approach would be to calculate the stringency relations between W-prefering constraints. This is the specific-over-general faithfulness bias that Prince and
Tesar (2004) consider, and which they point out will be unable to find the right constraint to install in the Azba case. As just emphasized above: looking just at the W-prefering constraints, the learner will not be required by an F-stringency bias to rule out Ident(z), because the specific Ident(z)-Onset constraint doesn’t prefer any winners:

104) BCD learning of Azba with W-prefering stringent constraint bias

<table>
<thead>
<tr>
<th></th>
<th>Agree (voice)</th>
<th>*b</th>
<th>*z</th>
<th>Ident(b)-Onset</th>
<th>Ident(z)-Onset</th>
<th>Ident(b)</th>
<th>Ident(z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) ab – ap</td>
<td>e</td>
<td>L</td>
<td>e</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td></td>
</tr>
<tr>
<td>(ii) ba – pa</td>
<td>e</td>
<td>L</td>
<td>e</td>
<td>W</td>
<td>e</td>
<td>W</td>
<td>e</td>
</tr>
<tr>
<td>(iii) azba – aspa</td>
<td>e</td>
<td>L</td>
<td>L</td>
<td>W</td>
<td>e</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

To build stratum 1
Step 1: Install all Markedness constraints that prefer no Ls
Resulting stratum 1: Agree(voice)

To build stratum 2
Step 1: Install all Markedness constraints that prefer no Ls
Resulting stratum 2: -- empty --

Step 2: Find the set of W-prefering Faithfulness constraints
Constraints ruled out: Ident(z)
Remaining constraints: Id(b)-Onset, Id(b), Id(z)-Onset

Step 3: Rule out all unranked W-prefering Faithfulness constraints that are more stringent than any other.
Constraints ruled out: Id(b)
Remaining constraints: Id(b)-Onset, Id(z)

7.2.4 Summarizing the Azba results

The Azba discussion above suggests that there are two good ways of calculating specific-to-general relations among IO-faith constraints: either we calculate across context arguments for a small(er) set of constraints as I have done, or we calculate across a large(r) set of constraints as Hayes does. As I mentioned above, the approach that I use does have the potential benefit of requiring fewer calculations among contexts with potential contingent subset relations. But the real choice between these options (and any others) will have to be made by using them in learning algorithms to handle a wide variety of data.

7.3 Returning to Anti-Paninian rankings and phonotactic learning

The Azba example has brought out one further point about the workings of the specific-over-general faithfulness bias adopted here. What we’ve seen is that even when a general faithfulness constraint must be ranked above a markedness constraint, my learner will choose first to install the specific one, and then the general one. We can see this by focusing our attention on just the stop voicing constraints in the Azba example, in the Support and the final ranking:

105) The Support, repeated in part

<table>
<thead>
<tr>
<th></th>
<th>*b</th>
<th>Ident(b)-Onset</th>
<th>Ident(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) ab – ap</td>
<td>L</td>
<td>e</td>
<td>W</td>
</tr>
<tr>
<td>(ii) ba – pa</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
<tr>
<td>(iii) azba – aspa</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
</tbody>
</table>

106) The final ranking learned, repeated in part from 101:
Ident(b)-Onset >> Ident(b) >> *b

It is clear from the first error 105i) that general Ident(b) will have to be ranked above *b in the final ranking – and installing this faithfulness constraint would be
sufficient. In other words, our learner would get the right surface pattern if he had instead used a ranking algorithm that chose this ranking:

107) A different ranking for Azba, that is equally surface-restrictive:
\[ \text{Ident}(b) \gg *b \gg \text{Ident}(b)\text{-Onset} \]

Is there any problem with choosing 106) instead of 107) from this Support? One point already addressed in section 4.1.1 is that the ranking in 105) is sort of Anti-Paninian (Prince 1997 et seq.), in that it ranks a general faith constraint above a specific one with some constraint intervening.38 So to rephrase the question: is there any problem with a learner who never adopts Anti-Paninian rankings from this Support?

Luckily, no. The crucial thing about AP rankings among faithfulness constraints is that they can only be necessary to analyze alternations – and alternations will provide a Support different than the one in 105). Here I will demonstrate this using the Swedish voicing case from Lombardi (1999)39 as an example. I will not provide a full account of how the AP ranking should be learned, but merely suggest the aspects of the Support that would be relevant to its discovery.

From the perspective of phonotactic learning, the facts of Swedish voicing are just like Azba: coda-onset clusters can either be all voiced or voiceless. Using the by-now familiar constraints, the error in 108) shows only that some \text{Ident}[\text{voice}] constraint must out-rank \text{*voice} – so my BCD algorithm will choose the ranking in 109):

108) The Support for Swedish voicing in phonotactic learning (using the Lombardi, 1999 analysis)

\[
\begin{array}{cccc}
\text{Agree (voice)} & \text{*voice} & \text{Ident}(\text{vce})-\text{Onset} & \text{Ident}(\text{vce}) \\
\text{(i) azba} \sim \text{aspa} & \text{e} & \text{L} & \text{W} & \text{W} \\
\end{array}
\]

109) Resulting ranking
\[ \text{Agree[voice]} \gg \text{Ident}(\text{vce})\text{-Onset} \gg *\text{voice} \gg \text{Ident}(\text{vce}) \]

However, this ranking turns out not to be the right one for Swedish. What alternations from Swedish demonstrate is that obstruent clusters can only be voiced on the surface if \textit{both} of their input members were voiced, as in 110d). If either input obstruent was voiceless, the cluster will also be voiceless (110a-c):

110) The truth about Swedish – from alternations

\[
\begin{array}{ccc}
\text{(a) /apta/} & \rightarrow & \text{[apta]} \\
\text{(b) /apda/} & \rightarrow & \text{[apta]} \\
\text{(c) /abta/} & \rightarrow & \text{[apta]} \\
\text{(d) /abda/} & \rightarrow & \text{[abda]} \\
\end{array}
\]

The ranking that the phonotactic learner in 109) adopted, however, cannot explain the mapping in (110b): the fact that /apda/ devoices to [apta], even though its input onset member was voiced. Thus, assuming this ranking will lead to the following error:

111) The error made by the phonotactic learning grammar:

\[
\begin{array}{cccc}
\text{Agree (voice)} & \text{*voice} & \text{Ident}(\text{vce})-\text{Onset} & \text{Ident}(\text{vce}) \\
\text{/apda/} & \text{\#} & \text{\*} & \text{\*} \\
\text{\# apa} & \text{\*} & \text{\*} & \text{\*} \\
\end{array}
\]

---

38 Though it is not crucial that the general ranks above the specific – thus, it might be better to refer to these rankings as ‘incidentally Anti-Paninian’.
39 This discussion owes much to Alan Prince’s LSA 2005 summer institute course notes.
The Support for Swedish voicing, including alternations

<table>
<thead>
<tr>
<th>input</th>
<th>winner – loser</th>
<th>Agree (voice)</th>
<th>*voice</th>
<th>Ident(vce)-Onset</th>
<th>Ident(vce)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(i) /azba/</td>
<td>azba – aspa</td>
<td>e</td>
<td>L</td>
<td>W</td>
<td>W</td>
</tr>
<tr>
<td>(ii) /apda/</td>
<td>apta – abda</td>
<td>e</td>
<td>W</td>
<td>L</td>
<td>W</td>
</tr>
</tbody>
</table>

At this point, our learner has evidence that Ident(vce)-Onset chooses a loser. This is something novel about learning from alternations – recall that in phonotactic learning, IO-faithfulness could only assign es and Ws. In the face of L-preferring faithfulness constraints, the F-specificity bias that I’ve suggested could be amended. For example, Step 3 could install faithfulness constraints that prefer some winners and no losers – this would correctly choose the general Ident(voice) constraint in 112).

Whatever the right strategy to deal with alternations, my present point is only that general >> specific rankings among faithfulness constraints cannot be necessary on the basis of phonotactics alone. Since the learner must already be assuming unfaithful I-O mappings before they’re crucial, this investigation will fall into the broad category of issues for restrictive learning in the face of alternations.

7.4 Summary and outstanding issues

In this section I have implemented my bias for the most specific IO-faithfulness constraints possible, using the tools I built in section 6 for calculating both universal and contingent specific-to-general relations between the contexts of faithfulness constraints. I have also provided some discussion of how this bias compares to the ones considered by Hayes (2004) and Prince and Tesar (2004), and raised the issue of how this bias will carry over to the learning of alternations (§7.3.)

There are of course many outstanding issues. One central issue is how the IO-faith specification bias interacts with other principles that enforce restrictiveness in the ranking of IO-faith constraints – that is, whether the details of Step 2c) of my algorithm are the best ones. But remaining questions should also be asked about the workings of the bias itself. One already alluded to in footnote 33 is the potential interaction between degrees of specificity in prosodic and subsegmental contexts. That is: what should the BCD learner do when choosing between the following two constraints?

Two potential faithfulness constraints with conflicting specificities

a) Ident[voice]-Stop
b) Ident[voice]-Obstruent-Onset

The problem is that stops are more specific than obstruents, but onsets are more specific than segments. Thus, at the subsegmental level 113a)’s context is more specific, but at the prosodic level 113b)’s context is more specific. In such a case – which constraint should the learner install? A second issue is the possibility that multiple faithfulness constraints might need to be considered to determine the specific-to-general relations between context arguments.

In part, the answer to these questions will come from a better understanding of the correct theory of faithfulness. (In the first instance: if there is no constraint like Ident[voice]-Obstruent-Onset, that particular ranking indeterminacy will never arise.)

8. Chapter 2 Summary, in preparation for Chapter 3

This chapter has presented a view of error-driven phonotactic learning in Optimality Theory, using one of a class of Biased Constraint Demotion algorithms. I
have drawn together arguments from the literature to support three ranking biases for choosing the most restrictive grammar consistent with learning data. Together, these biases aim to rank OO-faith above Markedness, and Markedness above IO-Faith, and to install only the most specific of IO-Faithfulness constraints rather than any more general ones. I have focused in particular on this third bias, and argued that the difficulties in determining the most specific faithfulness constraints should be handled by calculating specificity across the contexts of faithfulness and examining the relations between those contexts in the language’s observed output forms (winners.) And in discussing the benefits of BCD learning, I have continually stressed the use of stored errors – the Support – to ensure that restrictive grammars can be learned at every stage of acquisition, and so that early errors about the learning data do not persist in later rankings.

Most of the rest of this dissertation is concerned with how this BCD-style learner can be used to describe and predict aspects of natural language acquisition. Before embarking on this project, however, I note that Prince and Tesar are very explicit about the limited connection between their work on restrictiveness and the analysis of child data: 114) “It is important, however, to keep the subset issue notionally distinct from issues in the analysis of early acquisition patterns. The proposals we shall entertain are not intended to provide a direct account for child language data, although we expect that they ought to bear on the problem in various ways.” (Prince and Tesar 2004: 250)

With this caveat in mind – what aspects of child language data could this BCD learner speak to?

One prediction is that the biases of BCD characterize what is usually called the ‘initial state’. In other words, they provide the ranking of constraints before any errors have been made and ERC rows added to the Support.40 Thus, they also provide the grammar which begins the process of creating errors. For the BCD learner I gave in section 7.1, this initial ranking will look like 115):

115) The initial ranking, chosen by my BCD algorithm in the absence of ERC rows

OO-Faith >> Markedness >> IO-Faith

Note that the IO-faith specificity bias can’t have any initial ranking effect; this is because IO-faithfulness constraints are only spread out in the ranking if errors demand it, and until then are dumped in one stratum at the bottom of the hierarchy by Step 4.

What does this ranking tell us about early grammars? The high Markedness bias predicts that early grammars will be unmarked compared to the target. The high OO-faith bias makes predictions only once morphology has been learned; I will return to some support for this prediction in chapter 4 with data from Kazazis (1969), Bernhardt and Stemberger (1998) and Smith (1973). And while the IO-faith specificity bias isn’t relevant here, the existence of both specific and general faithfulness constraints at the bottom of our hierarchy predicts repairs in unprivileged contexts.

But what about stages of BCD learning once errors have been made? This is the subject of chapter 2.

40 Shelley Velleman (p.c.) points out that this pure initial state is clearly not the state of the grammar at the advent of phonological production or word learning. By the time they begin to talk, or even to produce canonical babble, children have learned many language-specific aspects of their phonological grammar. I remain agnostic as to the point at which this knowledge is rightly represented using an OT grammar of the sort being assumed here, but it will at least be before meaningful speech production begins.