Chapter 5

Learning biases and optimal learning conditions

Simulating acquisition of a restrictive grammar with an algorithmic learner involves many potential variables, parameters, and biases. In this chapter I introduce those factors that are relevant to learning using the constraint set proposed in Chapter 3, and discuss the impact that each has on learning the phonotactics of the vowel patterns in the sample languages North Estonian, Finnish, and North Seto. In particular, I present implementations of two biases that are entirely novel to this learning context: (a) a persistent bias that prioritizes specific over general faithfulness constraints, adapted from Hayes's (2004) *Favour Specificity* principle (originally proposed for a batch learner) and (b) several versions of an initial bias that prioritizes general over specific markedness constraints, one in particular adapted from Albright and Hayes (2006).

In Section 5.1 I introduce the factors that are assumed to remain constant, present learning results given these foundational assumptions, and discuss challenges to be overcome from this starting point. Although there are a number of different obstacles that contribute to the difficulty of learning these languages in the context of their typology, they are not all immediately obvious. Therefore each of Sections 5.2, 5.3, and 5.4 introduces a particular learning bias that is applied in order to address an already-identified problem, while simultaneously uncovering more subtle obstacles not previously apparent. Section 5.2 investigates the implementation of the specific-over-general faithfulness bias, which facilitates privileging first-syllable vowels but reveals that faithfulness constraints are nevertheless promoted too high. Section 5.3 focuses on options for varying the promotion rate applied at each learning update, which tempers the dramatic climb of the faithfulness constraints but shows that overly specific markedness generalizations are being learned in some cases. Section 5.4 explores the implementation of the general-over-specific markedness bias, which prioritizes more restrictive (more general) markedness constraints over less restrictive ones. Finally, in Section 5.5, I summarize all of the learning simulations performed with various combinations of values for the biases introduced in the preceding sections and generalize a set of ideal conditions for learning these Finnic languages.

For reference, Figures 5.1, 5.2, and 5.3 reproduce the Hasse diagrams representing target grammars for North Estonian, Finnish, and North Seto originally depicted in Figures 3.2, 3.3, and 3.4. Each node (whether containing just one constraint or a group of constraints) is assigned a particular colour so that the final rankings in the learning results that follow can be easily compared to the target grammars. Recall that the constraints in the grey nodes are not crucially ranked with respect to any of the others. Readers may refer to Figure 3.1 to review in more detail the general structure and colour-coding schema used in the Hasse diagrams for all three languages.

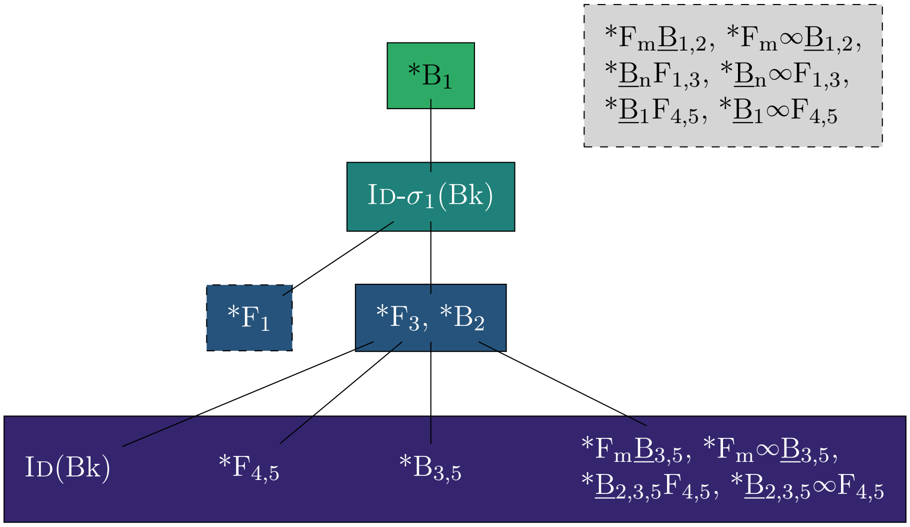

$*F_m\underline{B}_{1,2}$, $*F_m\infty\underline{B}_{1,2}$,
$*\underline{B}_nF_{1,3}$, $*\underline{B}_n\infty F_{1,3}$,
$*\underline{B}_1F_{4,5}$, $*\underline{B}_1\infty F_{4,5}$
$*B_1$
Id-σ_1(Bk)
$*F_1$
$*F_3$, $*B_2$
Id(Bk)
$*F_{4,5}$
$*B_{3,5}$
$*F_m\underline{B}_{3,5}$, $*F_m\infty\underline{B}_{3,5}$,
$*\underline{B}_{2,3,5}F_{4,5}$, $*\underline{B}_{2,3,5}\infty F_{4,5}$

Figure 5.1: Overall North Estonian rankings

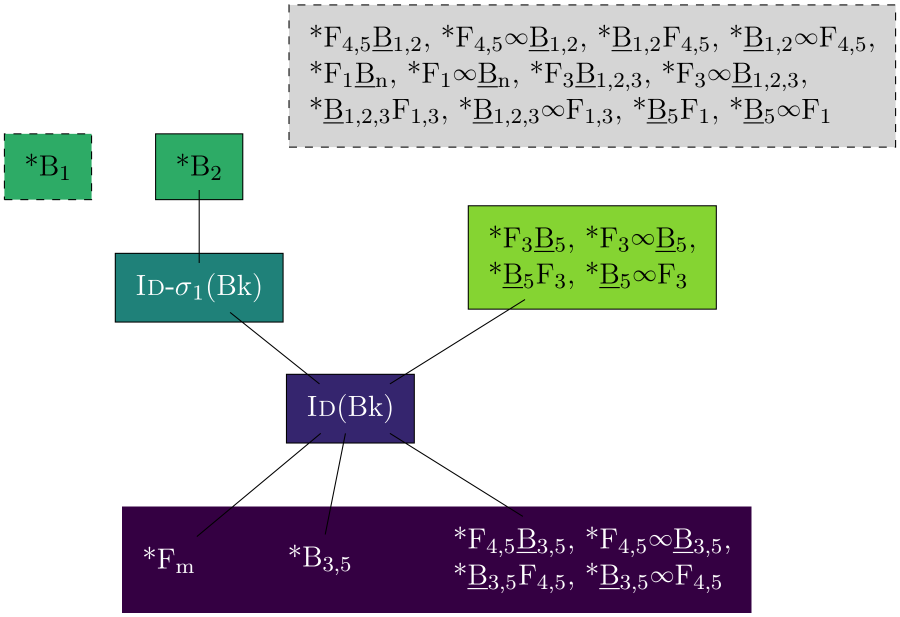

$*F_{4,5}\underline{B}_{1,2}$, $*F_{4,5}\infty\underline{B}_{1,2}$, $*\underline{B}_{1,2}F_{4,5}$, $*\underline{B}_{1,2}\infty F_{4,5}$,
$*F_1\underline{B}_n$, $*F_1\infty\underline{B}_n$, $*F_3\underline{B}_{1,2,3}$, $*F_3\infty\underline{B}_{1,2,3}$,
$*\underline{B}_{1,2,3}F_{1,3}$, $*\underline{B}_{1,2,3}\infty F_{1,3}$, $*\underline{B}_5F_1$, $*\underline{B}_5\infty F_1$
$*B_1$
$*B_2$
$*F_3\underline{B}_5$, $*F_3\infty\underline{B}_5$,
$*\underline{B}_5F_3$, $*\underline{B}_5\infty F_3$
Id-σ_1(Bk)
Id(Bk)
$*F_m$
$*B_{3,5}$
$*F_{4,5}\underline{B}_{3,5}$, $*F_{4,5}\infty\underline{B}_{3,5}$,
$*\underline{B}_{3,5}F_{4,5}$, $*\underline{B}_{3,5}\infty F_{4,5}$

Figure 5.2: Overall Finnish rankings

5.1 Learning simulations with default settings

The Gradual Learning Algorithm (GLA), as specified by Boersma and Hayes (2001) and discussed in Sections 4.1.2 and 4.2, describes the general procedure for this type of gradual, error-driven learning. The bare bones of the learning algorithm as described lay the foundation for additional potential parameters or biases to be included.

5.1.1 Learning parameters/biases assumed to remain constant

For the purposes of this project, there are a number of parameters that I considered allowing to vary, but ultimately decided to keep constant. These parameters, detailed below, are: initial markedness

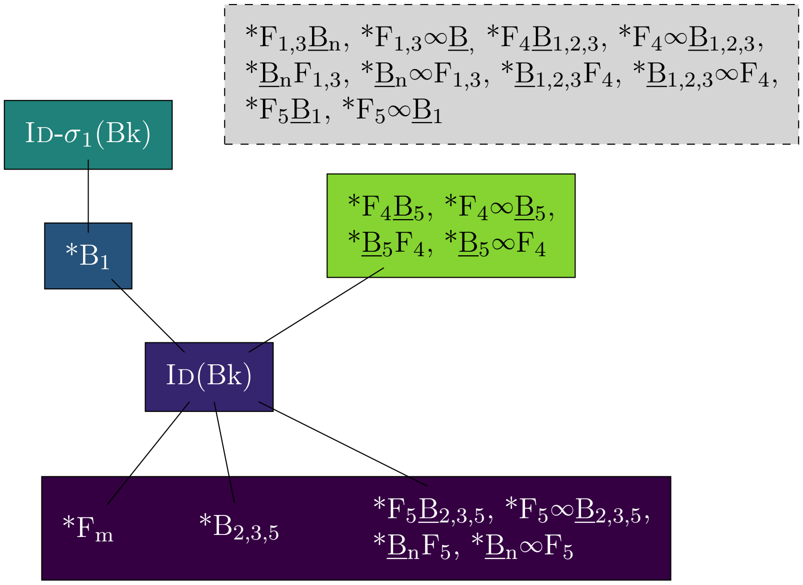

$*F_{1,3}\underline{B}_{n}$, $*F_{1,3}{\infty}\underline{B}$, $*F_{4}\underline{B}_{1,2,3}$, $*F_{4}{\infty}\underline{B}_{1,2,3}$, $*\underline{B}_{n}F_{1,3}$, $*\underline{B}_{n}{\infty}F_{1,3}$, $*\underline{B}_{1,2,3}F_{4}$, $*\underline{B}_{1,2,3}{\infty}F_{4}$, $*F_{5}\underline{B}_{1}$, $*F_{5}{\infty}\underline{B}_{1}$
Id-σ_1(Bk)
$*B_1$
$*F_{4}\underline{B}_{5}$, $*F_{4}{\infty}\underline{B}_{5}$, $*\underline{B}_{5}F_{4}$, $*\underline{B}_{5}{\infty}F_{4}$
Id(Bk)
$*F_m$
$*B_{2,3,5}$
$*F_{5}\underline{B}_{2,3,5}$, $*F_{5}{\infty}\underline{B}_{2,3,5}$, $*\underline{B}_{n}F_{5}$, $*\underline{B}_{n}{\infty}F_{5}$

Figure 5.3: Overall North Seto rankings

over faithfulness, demoting all loser-preferring constraints, permitting negative ranking values, and the organization of the learning simulation into four stages with a total of 20 000 learning trials.

The first of these determines whether all constraints have the same initial ranking values or if faithfulness constraints should start lower than markedness constraints. I consistently apply a *low-faithfulness* bias in these simulations.[40] The bias toward low initial faithfulness is widely used in the learning literature, as it helps to ensure that the acquired grammar is as restrictive as possible; that is, it mitigates the Subset Problem (Angluin, 1980; Baker, 1979). Readers can find more detailed discussion in, e.g., Gnanadesikan (1995); Hayes (2004); Jesney and Tessier (2011); Prince and Tesar (2004); Smolensky (1996). The default implementation of this bias in this project is to set the initial ranking value of faithfulness constraints to be 0, and that of markedness constraints to be 100. There are some other biases discussed in later parts of this chapter that will set initial markedness values to be different from the default; however, these will continue to preserve the overarching low-faithfulness bias.

The second parameter that will remain constant is that of demotion eligibility; that is, whether *all* loser-preferring constraints get demoted at each learning update, or just the *undominated* ones. In all learning simulations, I demote all loser-preferring constraints rather than choosing to run some simulations in which only the undominated ones get demoted. Boersma and Hayes (2001) find that demoting only undominated loser-preferrers cause the GLA to fail on their test data. On the other hand, Magri (2012) shows that demoting all losers can prevent the learner from converging efficiently. Suffice it to say that even if choosing to demote all loser-preferrers may affect the learner's ability to converge *efficiently*, it will not affect whether or not the learner converges *at all*.

The third constant is the number of learning trials, which is fixed at 20 000 for each simulation. All simulations described herein converged well before iterating through this many trials, providing a long enough timeline to ensure that even the odd later error (caused by a particularly noisy

[40] I also ran a small number of exploratory simulations in which faithfulness constraints experience a more persistent downward bias, being demoted at regular intervals through the learning process. However, these experiments did not produce any promising results so I set the notion of "gravity" aside and did not pursue it any further.

evaluation) did not affect the overall ranking.

The fourth parameter that will remain constant is permitting constraints to take on negative ranking values. It is possible for a constraint to prefer a loser and therefore be eligible for demotion even if it already has a ranking value of 0. In some learning situations a drop into negative ranking values is not permitted. However, since I am working with ranked (classic OT) rather than weighted (e.g. Harmonic Grammar) constraints, there is no particular concern associated with negative ranking values; all ranking values are converted to relative ordinal rankings at evaluation so the actual numerical values themselves are irrelevant. For example, the values $\{\theta(C_1) = 100,\ \theta(C_2) = 50\}$ produce the exact same ranking as the values $\{\theta(C_1) = -25,\ \theta(C_2) = -32\}$. Given this fact, the default OTSoft (Hayes et al., 2013) approach for GLA learning is used; that is, to permit demotion of constraints even when the resulting ranking value is negative.

The last few parameters that are held constant across simulations are the organization of learning trials into stages, as well as evaluation noise and the plasticity function. Recall that *noise* is the standard deviation of the normal distribution centred at each constraint's ranking value (used for perturbing the ranking values at evaluation time) and that *plasticity* is the amount added to or subtracted from each constraint's value during an update. Detailed definitions of each are in Section 4.2. Based on pilot simulations run early on in this project, none of the results I discuss appear to depend on changes to these settings, so I use the default OTSoft (Hayes et al., 2013) assumptions for noise, evenly-distributed learning trials, and decreasing plasticity in GLA learning; they are summarized in Table 5.1.

Parameter	**Stage 1**	**Stage 2**	**Stage 3**	**Stage 4**
Number of learning trials	5 000	5 000	5 000	5 000
Evaluation noise	2	2	2	2
Plasticity	2	0.2	0.02	0.002

Table 5.1: Invariant settings for GLA learning.

5.1.2 Simulation results - default parameters

Here and throughout the rest of this chapter, all sections presenting simulation results will follow the same general structure. First, I define the parameter settings for the learner. Next, I summarize in a table the average success of the learner on each of the three sample languages. Finally, I present detailed results and discuss final rankings for each of the individual languages.

Note also that for each new parameter introduced throughout the chapter, I show an example of its application first on its own, then in combination with previous parameters. Some settings for each parameter produce better results than others, of course, but for the purpose of illustrating each parameter's effect without getting bogged down with granular details I select just one setting for each option. Once all of the parameters have been defined, Section 5.5 considers the results from the full crossing of parameter settings.

Initial learning simulations use all of the basic parameters (those described in Section 5.1.1) at their default settings, with no additional biases or parameters introduced. Learner A is defined with the settings in Table 5.2.

Under these default conditions, learning simulations for all three sample languages fail to acquire the target grammars, producing instead fully-faithful grammars. Such grammars succeed 100% of

Learner A:	Parameter	Setting
	All basic parameters	Default • low initial faithfulness (0) / high markedness (100) • demote *all* loser-preferring constraints • permit negative ranking values • evaluation noise = 2 • 20 000 learning trials per simulation, split into 4 stages • plasticity = $2, 0.2, 0.02, 0.002$ through Stages 1 to 4, respectively

Table 5.2: Parameter settings for Learner A.

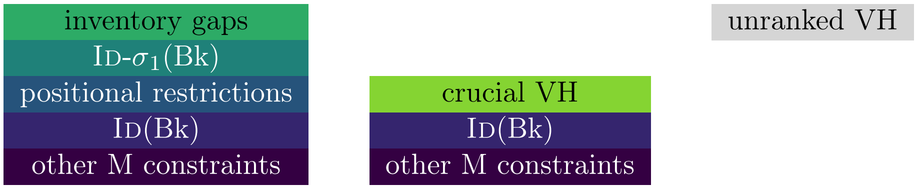
the time on the input forms, since I assume idempotence. However, they are very poorly equipped to deal with illicit test forms (such as the "excluded sequences" shown in Tables 4.1, 4.2, and 4.3). Test results are summarized in Table 5.3. Here, and throughout the rest of the chapter, these results are calculated as per the evaluation procedure described in Section 4.4.1.[41] The tests are performed with inputs comprising both possible and impossible vowel sequences in each language. Results and final rankings for each individual language are discussed in more detail below.

Language	Average rate of correct outputs (%)
North Estonian	24.55%
Finnish	26.83%
North Seto	29.84%

Table 5.3: Summary of results from simulations with Learner A.

Below and throughout the rest of this chapter, I present the final ranking values for selected constraints in each language, in order to illustrate the particular successes or problems of each learning simulation without overwhelming the reader with all seventy-four constraint values. As a reminder, in each of these tables, the same colour schemes are used as in the Hasse diagrams in Figures 5.1, 5.2, and 5.3.

(112) The desired relative orderings are always as shown:



In addition, I separate individual rows of the tables using one of three options (the probabilities given here are from Table 4.5):

- A heavy solid line between rows means that the constraint values are separated by at least 12 and are therefore at least 99.999% likely to maintain the same ranking relationship after evaluation noise is applied.

[41] Recall that any results better than approximately 0.9 are based on the evaluations of $n = 100$ tests, whereas those below 0.9 are based on only $n = 5$ tests.

- A double dashed line between rows means that the constraint values are separated by at least 4 and are therefore at least 92.135% likely to maintain the same ranking relationship after evaluation noise is applied.
- No line between rows means that the constraint values are separated by less than 4 and are therefore less than 92.135% likely to maintain the same ranking relationship after evaluation noise is applied.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.4. Id(Bk) has risen to the top of the rankings, producing a grammar that does not align with the crucial rankings proposed in Section 3.2.2. The target rankings require:

$*B_1$ >> Id-σ_1(Bk) >> $*F_3$, $*B_2$ >> Id(Bk)

Constraint	Final ranking value
Id(Bk)	116
$*\underline{B}_5F_3$	104
$*\underline{B}_5\infty F_3$	104
$*B_1$	104
$*F_3$	102
$*B_2$	102
$*F_5\underline{B}_2$	100
$*F_5\infty\underline{B}_2$	100
Id-σ_1(Bk)	92

Table 5.4: Excerpt of final ranking values for North Estonian after simulation with Learner A.

Once general faithfulness is highest-ranked, no further learner trials will cause any errors. Due to the fact that these learners receive positive evidence only, the fully-faithful candidate will always be the intended winner and no further updates will be made to the grammar.

Finnish: Table 5.5 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. Id(Bk) is at the top of this grammar too, meaning that it does not achieve the crucial rankings proposed in Section 3.2.3. The target rankings require:

$*B_2$ >> Id-σ_1(Bk) >> Id(Bk)

$*F_3\underline{B}_5$, $*F_3\infty\underline{B}_5$, $*\underline{B}_5F_3$, $*\underline{B}_5\infty F_3$ >> Id(Bk)

North Seto: Learning from simulated North Seto data results in final ranking values for a selection of crucial constraints shown in Table 5.6. Id(Bk) has once again risen to the top; the resulting grammar does not meet the crucial target rankings proposed in Section 3.2.4. The target rankings require:

Id-σ_1(Bk) >> $*B_1$ >> Id(Bk)

$*F_4\underline{B}_5$, $*F_4\infty\underline{B}_5$, $*\underline{B}_5F_4$, $*\underline{B}_5\infty F_4$ >> Id(Bk)

Constraint	Final ranking value
Id(Bk)	116
*B_2	110
*$F_3\infty\underline{B}_5$	106
*$\underline{B}_5\infty F_3$	106
*$F_3\underline{B}_5$	104
*$\underline{B}_5F_3$	104
Id-σ_1(Bk)	80

Table 5.5: Excerpt of final ranking values for Finnish after simulation with Learner A.

Constraint	Final ranking value
Id(Bk)	116
*$F_4\underline{B}_5$	106
*$F_4\infty\underline{B}_5$	106
*$\underline{B}_5\infty F_4$	106
*B_1	102
*$\underline{B}_5F_4$	100
Id-σ_1(Bk)	96

Table 5.6: Excerpt of final ranking values for North Seto after simulation with Learner A.

5.1.3 Discussion

There are several obstacles that must be addressed on the way to acquiring better – even excellent – final grammars. However, in the results shown above in Section 5.1.2, not all of the challenges are apparent; some only become clear as the initial problems are resolved. In this section I discuss those that are immediately identifiable, and leave the others to be discussed and addressed in subsequent sections of this chapter.

With respect to the results presented in Section 5.1.2, the most glaring problem is that Id(Bk) is highest ranked in all three grammars. This means that during the learning process, Id(Bk) rises all the way from its initial value of 0, past all of the markedness constraints starting at 100, to the very top of the rankings. Such grammars are fully faithful and therefore overgenerate to the point of excluding nothing. That is, any vowel is permitted anywhere in the word (whether in an initial syllable or not), and any co-occurrence of vowels (whether harmonic or disharmonic) is likewise permitted, since there are no markedness constraints above Id(Bk) to enact any pressure to the contrary. In other words, we see none of the desired features of of these languages: inventory gaps, positional neutralization, or harmony.

The reason for Id(Bk)'s rise all the way to the top of the rankings is that there is no obligation (or even tendency) for Id-σ_1(Bk) to outrank Id(Bk), which is problematic particularly under assumption of idempotence. Each time the learner encounters an error, Id(Bk) is always a winner-preferring constraint, since the underlying form is assumed to be identical to the heard surface form. The Elementary Ranking Conditions (ERC) matrix in (113) shows that for an observed form such as /o...ɑ/ (grammatical in all three of the sample languages), Id(Bk) is a winner-preferrer for *any* error and ID-σ_1(Bk) is a winner-preferrer for only those errors involving the first syllable. Thus an error in the first syllable will result in promotion of both faithfulness constraints, but any error past the first syllable will result in promotion of only the general one. Since a crucial element of all three sample languages' target grammars is for ID-σ_1(Bk) to outrank ID(Bk), this ranking will never be achieved and the learner will only stop making errors once ID(Bk) has been promoted all the way to the top of the rankings.

(113) ERC matrix demonstrating that under assumption of idempotence, all learning errors have ID(Bk) as a winner-preferring constraint.

input	*winner ∼ loser*	Markedness constraints	ID(Bk)	ID-σ_1(Bk)
/o...ɑ/	o...ɑ ∼ o...æ	...	W	e
/o...ɑ/	o...ɑ ∼ ø...ɑ	...	W	W
/o...ɑ/	o...ɑ ∼ ø...æ	...	W	W

Addressing the relative ranking of specific vs general faithfulness constraints is not the only obstacle to successful learning of grammars for the sample languages. However, as it is the only one apparent under the learning conditions presented in Section 5.1, it must be addressed before any others can be revealed. Section 5.2 presents two possible solutions to this problem.

5.2 $\mathrm{F}_{\mathrm{spec}} >> \mathrm{F}_{\mathrm{gen}}$ bias

The constraint set that I use for this project includes only two faithfulness constraints, ID(Bk) and ID-σ_1(Bk), the first applying more broadly and the second in a narrower context. When two such versions of a faithfulness constraint exist, it is possible to construct a grammar in which marked elements in underlying forms surface only in privileged contexts. For example, recall the constraint *F$_3$, which is violated by vowels in set F$_3$ = {æ, ø, y}. Then the ranking

ID-σ_1(Bk) >> *F$_3$ >> ID(Bk)

bans vowels in set F$_3$ in general, but permits them in initial syllables.

A specific-over-general faithfulness bias ($\mathrm{F}_{\mathrm{spec}} >> \mathrm{F}_{\mathrm{gen}}$) is a strategy that can help find the most restrictive grammar that accounts for the input data, avoiding a superset (overgenerating) grammar (Hayes, 2004; Tessier, 2007). I explore two slightly different approaches to this idea: one is to define a set minimum distance by which the specific faithfulness constraint must exceed the general one (referred to as the *a priori* bias), and the other is to prioritize promotion of the specific constraint even when the general one could also be promoted (referred to as the Favour Specificity bias). These two approaches are detailed in Sections 5.2.1 and 5.2.2, respectively, and Section 5.2.3 discusses the advantage of both being used together.

5.2.1 *A priori* bias

One approach to the specific-over-general faithfulness bias is to ensure that the ranking value for the specific version of the constraint is a minimum specified distance higher than that of the general version. The satisfaction of this bias is enforced persistently through the learning simulation, both in the initial state and after each individual learning update.

5.2.1.1 Rationale

Maintaining a minimum difference between the ranking values of a specific-general pair of faithfulness constraints ensures that the specific version of the constraint always has a better opportunity to claim credit for a particular output form than the general one does, corresponding to a more restrictive grammar overall.

5.2.1.2 Implementation

The $F_{spec} >> F_{gen}$ bias between any specific-general pair of faithfulness constraints can be implemented by means of an *a priori* bias that ensures $\theta(F_{spec}) - \theta(F_{gen}) \geq d$, for some distance d. Practically, the learner adjusts the initial ranking values such that any two constraints in this type of relationship are at least d apart, and then does the same after each learning update. I propose that if the two constraints have a difference of less than d, then it is always the case that the specific one has its value increased rather than the general one having its value decreased, as the latter means that faithfulness constraints are only ever promoted when both are violated. OTSoft (Hayes et al., 2013) sets the default value of this difference to be $d = 20$, stating that it is "very close probabilistically to being an obligatory ranking" (Hayes, 2013, p. 24).

In my learning simulations, I test the omission of this bias as well as a range of different d values: 0 (i.e., $\theta(F_{spec})$ must be no less than $\theta(F_{gen})$), 10, 20, 30, and 40.

5.2.1.3 Simulation results - *a priori* bias

To demonstrate the effect of the *a priori* bias, I simulate acquisition of the three sample languages using Learner B, defined with the settings in Table 5.7. The selection of $d = 20$ for illustrative purposes is drawn from the OTSoft default as mentioned above. Results using learners with other values of d are summarized in Section 5.5.

Learner B:	**Parameter**	**Setting**
	All basic parameters	Default
	A priori bias ($F_{spec} >> F_{gen}$)	$d = 20$

Table 5.7: Parameter settings for Learner B.

With the *a priori* bias set to $d = 20$, learning simulations for all three sample languages still fail to acquire the target grammars. The learner trained on North Estonian data produces a grammar that, while not correct, does have some promising characteristics. On the other hand, the learners trained on Finnish and North Seto data once again produce fully-faithful grammars. Test results are summarized in Table 5.8; results and final rankings for each individual language are discussed in more detail below.

Language	**Average rate of correct outputs (%)**
North Estonian	78.63%
Finnish	28.95%
North Seto	31.33%

Table 5.8: Summary of results from simulations with Learner B.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.9. Several of the crucial relative rankings

$*B_1$ >> Id-σ_1(Bk) >> $*F_3$, $*B_2$ >> Id(Bk)

proposed in Section 3.2.2, are met by this grammar. However, one of the key elements – the B_1 inventory gap – is missing, by virtue of that fact that the final value of $*B_1$ is not only not at the top, but below even Id(Bk). Thus the acquired grammar will incorrectly permit the B_1 vowel /ɯ/ in initial syllables.

Constraint	**Final ranking value**
Id-σ_1(Bk)	129.22
$*B_2$	115.00
$*F_3$	110.22
Id(Bk)	109.22
$*\underline{B}_5\infty F_3$	108.00
$*B_1$	108.00
$*F_5\infty\underline{B}_2$	106.80
$*\underline{B}_5 F_3$	104.00
$*F_5\underline{B}_2$	98.18

Table 5.9: Excerpt of final ranking values for North Estonian after simulation with Learner B.

The ranking acquired by this learner does generally follow the required positional restrictions by ranking Id-σ_1(Bk) >> $*F_3$, $*B_2$ >> Id(Bk); however, the ranking values are close enough together that the stochastic nature of evaluation results in somewhat variable adherence to these positional restrictions. For example, the ungrammatical test input /y...æ/ would be expected to surface as [y...ɑ], neutralizing the restricted vowel in the second syllable. However, during testing, this grammar selects output candidates with the frequencies shown in Table 5.10.

Candidate	**Output frequency (%)**	
	Actual	**Desired**
y...æ	35	0
y...ɑ	65	100
u...æ	0	0
u...ɑ	0	0

Table 5.10: Frequency of candidate selection for input /y...æ/ with North Estonian grammar acquired by Learner B. Number of sample evaluations $n = 100$.

Although the ranking values of the top-ranked constraints are crowded quite close together (with the stochastic component of evaluation producing variable outputs as in Table 5.10), at least one success the North Estonian learner achieves is that the learner converges with markedness constraints between the two faithfulness constraints.

Finnish: Table 5.11 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. Both faithfulness constraints have risen to the top. The distance of Id(Bk) above $*B_2$ and the relevant no-disagreement constraints is small enough that evaluation noise might cause it to swap rankings with one of its neighbours. However, in order to meet the crucial rankings

$*B_2$ >> Id-σ_1(Bk) >> Id(Bk)

$*F_3\underline{B}_5$, $*F_3\infty\underline{B}_5$, $*\underline{B}_5F_3$, $*\underline{B}_5\infty F_3$ >> Id(Bk)

proposed in Section 3.2.3, such swaps would have to be guaranteed to occur at every evaluation, which is extremely unlikely given the final ranking values. Hence the final grammar produced by Learner B on Finnish inputs is more or less fully faithful, since ranking both faithfulness constraints at the top without any markedness constraints in between is in practice indistinguishable from a grammar such as the one acquired by Learner A where Id(Bk) is top-ranked alone (with Id-σ_1(Bk) far below). A representative evaluation is shown in Tableau (114).

Constraint	**Final ranking value**
Id-σ_1(Bk)	136
Id(Bk)	116
$*\underline{B}_5\infty F_3$	112
$*F_3\underline{B}_5$	110
$*F_3\infty\underline{B}_5$	110
$*\underline{B}_5F_3$	110
$*B_2$	110

Table 5.11: Excerpt of final ranking values for Finnish after simulation with Learner B.

(114) Sample evaluation of test input /o...æ/ in the Finnish grammar acquired by Learner B. The grammar selects the faithful candidate [o...æ] as optimal even though it is not harmonic.

/o...æ/	Id-σ_1(Bk)	Id(Bk)	$*\underline{B}_5\infty F_3$	(e.g. $*B_2$, $*\underline{B}_5F_3$, $*F_3\underline{B}_5$, $*F_3\infty\underline{B}_5$, . . .)
☞ a. o...æ			*	
✓ b. o...ɑ		*!		
c. ø...æ	*!	*		
d. ø...ɑ	*!	**		

In principle it should have been reasonable for Id-σ_1(Bk) to end up with a final ranking value greater than or equal to the top-ranked markedness constraints with Id(Bk) lower down. However, at the time that Id-σ_1(Bk) approaches the highest-ranked markedness constraints (including $*B_2$ with $\theta(*B_2) = 110$), the other context-free markedness constraints all have values in $[100, 106]$ and are therefore within a small enough window for evaluation noise to make (e.g.) $*B_3$ or $*F_3$ active in selecting the optimal candidate. This results in errors and therefore more updates which push the faithfulness constraints ever higher. It is only once Id(Bk) surpasses this clump of markedness constraints that errors taper off and the learner converges.

Table 5.12 shows a select few stages in Finnish Learner B's learning trajectory highlighting the proximity of $*B_3$ and $*F_3$ to $*B_2$ as Id-σ_1(Bk) rises to the top. The *winner* ∼ *loser* notation at the top of each column indicates that with (e.g.) /e...ɑ...o/ as the input form, [e...ɑ...o] as the desired (faithful) output, and [e...ɑ...ø] as the optimal candidate selected by the learner's current hypothesized grammar, the learner has encountered an error and therefore must update its current grammar accordingly.

Trial #68		**Trial #72**		**Trial #89**	
e...ɑ...o ∼ e...ɑ...ø		y...y...ø ∼ y...u...o		æ...æ...i ∼ æ...ɑ...i	
$*B_2$	110	Id-σ_1(Bk)	112	Id-σ_1(Bk)	120
Id-σ_1(Bk)	108	$*B_2$	110	$*B_2$	110
$*B_3$	106	$*B_3$	108	$*\underline{B}_5\infty F_3$	108
$*F_3$	104	$*\underline{B}_5\infty F_3$	106	$*B_3$	106
$*B_1$	104	$*F_3$	104	$*F_3\underline{B}_5$	106
$*\underline{B}_5\infty F_3$	104	$*B_1$	104	$*F_3\infty\underline{B}_5$	106
$*F_1$	102	$*F_3\infty\underline{B}_3$	104	$*\underline{B}_5F_3$	106
$*F_3\underline{B}_3$	102	$*F_3\underline{B}_5$	104	$*F_3$	104
$*F_3\infty\underline{B}_3$	102	$*F_3\infty\underline{B}_5$	104	$*B_1$	104
$*F_3\underline{B}_5$	102	$*\underline{B}_5F_3$	104	$*F_3\infty\underline{B}_3$	104
$*F_3\infty\underline{B}_5$	102	$*F_3\underline{B}_3$	102	$*\underline{B}_5F_1$	104
$*\underline{B}_3\infty F_3$	102	$*F_4\infty\underline{B}_3$	102	$*\underline{B}_5\infty F_1$	104
$*\underline{B}_5F_1$	102	$*\underline{B}_3\infty F_3$	102	$*F_1$	102
$*\underline{B}_5\infty F_1$	102	$*\underline{B}_5F_1$	102	$*F_3\underline{B}_3$	102
$*\underline{B}_5F_3$	102	$*\underline{B}_5\infty F_1$	102	$*\underline{B}_3\infty F_3$	102
$*F_4$	100	$*F_1$	100	Id(Bk)	100
$*F_5$	100	$*F_4$	100	$*F_4$	100
$*B_5$	100	$*F_5$	100	$*F_5$	100
$*F_4\infty\underline{B}_3$	100	$*B_5$	100	$*B_5$	100
...	...	...	...	$*F_4\infty\underline{B}_3$	100
Id(Bk)	88	Id(Bk)	92	...	...

Table 5.12: The highest of Finnish Learner B's constraint ranking values after three different learning updates. Although the crucial constraints ($*B_2$, Id-σ_1(Bk), $*F_3\underline{B}_5$, $*F_3\infty\underline{B}_5$, $*\underline{B}_5F_3$, $*\underline{B}_5\infty F_3$) are in reasonably good positions, constraints such as $*F_1$, $*F_3$, $*F_4$, $*F_5$, $*B_3$, $*B_5$ are near enough to be potentially disruptive.

North Seto: Table 5.13 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. Similar to the Finnish results, both faithfulness constraints have risen to the top. Though the markedness constraints are correctly ordered relative to each other, the relative positions of the faithfulness vs the markedness constraints are not correct with respect to the crucial rankings proposed in Section 3.2.4. The target rankings require:

$$\text{Id-}\sigma_1\text{(Bk)} \gg *B_1 \gg \text{Id(Bk)}$$

$$*F_4\underline{B}_5,\ *F_4\infty\underline{B}_5,\ *\underline{B}_5F_4,\ *\underline{B}_5\infty F_4 \gg \text{Id(Bk)}$$

Again, the final grammar produced by Learner B on North Seto inputs is essentially fully faithful (due to both faithfulness constraints being at the top), with similar learning challenges as described for Finnish.

In the next section, I introduce a second possible approach to the $F_{spec} \gg F_{gen}$. Following that,

Constraint	Final ranking value
Id-σ_1(Bk)	136
Id(Bk)	116
$*F_4\infty\underline{B}_5$	110
$*\underline{B}_5\infty F_4$	110
$*F_4\underline{B}_5$	108
$*\underline{B}_5F_4$	108
$*B_1$	104

Table 5.13: Excerpt of final ranking values for North Seto after simulation with Learner B.

results from both Section 5.2.1 and Section 5.2.2 are discussed in Section 5.2.4.

5.2.2 Favour Specificity

This section proposes an alternative to the fixed, enforced *a priori* bias explored above. The underlying idea for the Favour Specificity bias is to allow the specific faithfulness constraint to rise independently of the general one, similar to the *Favour Specificity* principle that Hayes (2004) introduces for the Low-Faithfulness Constraint Demotion algorithm. Although that proposal focuses on a different algorithm, the same principle can be adapted to apply to the GLA as well.

5.2.2.1 Rationale

As discussed in 5.2.1, setting an *a priori* bias helps specific faithfulness constraints stay above their general counterparts. However, because each violation of a first-syllable faithfulness constraint is also necessarily a violation of a general faithfulness constraint, there is no opportunity for the specific constraint to ever rise any further above the general version than the *a priori* bias specifies. That is, it is always the case that either the pair of constraints is moving in tandem (if there is an error in the first syllable only) or the general constraint is "pushing" the specific one up from below (if there is at least one error in a non-initial syllable). Both of these scenarios have the same effect: the specific constraint does not ever move independently of the general one. Recall the ERC matrix in (113) for an illustration of this phenomenon.

This type of movement, where specific and general constraints are separated by what is effectively a constant distance, can cause a challenge for the learner in that the d value that is specified for the *a priori* bias may or may not be large enough for other necessary constraints and/or interactions to "fit" between the two faithfulness constraints, depending on the target grammar. For instance, suppose the target grammar has crucial rankings $M_1 \gg F_1 \gg F_2 \gg M_2$, and the learner is set to its task with a fixed difference (e.g., $d = 20$ between the two faithfulness constraints) assigned to the *a priori* bias. The $F_1 \gg F_2$ relationship will be effectively categorical, which is sufficient for this grammar. However, suppose the target grammar has instead crucial rankings $F_1 \gg M_1 \gg F_2 >$ $> M_2$. In this case, $d = 20$ does not create enough space: the constraints in either of the crucial rankings $F_1 \gg M_1$ or $M_1 \gg F_2$ (or both) will have ranking values close enough that evaluation noise will create some variability in surface forms. Conversely, attempting to solve this problem by arbitrarily setting the *a priori* bias to be larger can cause other issues instead (for example, it would prevent the learning of a target grammar where $F_1 \gg M_1 \gg F_2$ but M_1 must be variably interchangeable with both F_1 and F_2).

To address this challenge, I test an alternate approach (Favour Specificity) that allows the space between specific and general counterparts to change, depending on the kinds of errors that are made. Both the Favour Specificity and the *a priori* bias are ultimately combined in Section 5.2.3.

5.2.2.2 Implementation

As always, when a learning error triggers an update to the constraint ranking values, the relevant ERC is inspected for winner-preferring vs loser-preferring constraints. In this case, if both the specific and the general version of a particular faithfulness constraint are eligible for promotion (i.e., both prefer the winner), then only the specific one gets promoted. In (115a), Id-σ_1(Bk) does not prefer the winner so Id(Bk) is promoted. In (115b) and (115c), both Id(Bk) and Id-σ_1(Bk) prefer the winner and only Id-σ_1(Bk) is promoted.

(115) ERC matrix showing that Id(Bk) is promoted when and only when Id-σ_1(Bk) does not prefer the winner.

input	*winner ∼ loser*	Markedness constraints	Id(Bk)	Id-σ_1(Bk)
a. /o...ɑ/	o...ɑ ∼ o...æ	. . .	W ↑	e
b. /o...ɑ/	o...ɑ ∼ ø...ɑ	. . .	W $\not\uparrow$	W ↑
c. /o...ɑ/	o...ɑ ∼ ø...æ	. . .	W $\not\uparrow$	W ↑

There is also a set of optional variations to this implementation, in which the *a priori* bias (if any; see Section 5.2.1) increases if the current $\theta(\text{F}_{\text{spec}}) - \theta(\text{F}_{\text{gen}})$ difference is greater than d. However, pilot simulations testing these variations did not produce promising results, so they will not be discussed further. Rather, I focus on just the basic version of the Favour Specificity bias.

5.2.2.3 Simulation results - Favour Specificity

To demonstrate the effect of the Favour Specificity bias, I simulate acquisition of the three sample languages using Learner C, defined with the settings in Table 5.14.

Learner C:	**Parameter**	**Setting**
	All basic parameters	Default
	Favour Specificity bias (F_{spec} >> F_{gen})	Active

Table 5.14: Parameter settings for Learner C.

With the Favour Specificity bias applied, learning simulations for all three sample languages still fail to acquire the target grammars. Once again, the grammar acquired by the learner trained on North Estonian is a significant improvement over the one acquired by Learner A, but the Finnish and North Seto grammars are essentially fully faithful. Test results are summarized in Table 5.15, along with the final ranking values of Id-σ_1(Bk) and Id(Bk). Results and final rankings for each individual language are discussed in more detail below.

It is clear from the final values of Id(Bk) and Id-σ_1(Bk) that Favour Specificity has a much greater impact on North Estonian than on Finnish or North Seto; this benefit is discussed below. Additionally, based solely on its average rate of correct outputs, North Seto Learner C appears to have shown some improvement over Learner A. However, this is in fact a statistically convenient side effect of final ranking values that are no better from a theoretical perspective; further explanation is provided below.

Language	Average rate of correct outputs (%)	Final faithfulness ranking values
North Estonian	89.98%	Id-σ_1(Bk) 124; Id(Bk) 70
Finnish	26.96%	Id-σ_1(Bk) 116; Id(Bk) 112
North Seto	46.95%	Id-σ_1(Bk) 118; Id(Bk) 114

Table 5.15: Summary of results from simulations with Learner C.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.16. As for Learner B, several of the crucial relative rankings

*B_1 >> Id-σ_1(Bk) >> *F_3, *B_2 >> Id(Bk)

proposed in Section 3.2.2 are met by this grammar. The issue of the inventory gap is still relevant – *B_1 is still not at the top of the rankings – but at least it is above Id(Bk).

Constraint	Final ranking value
Id-σ_1(Bk)	124.00
*F_3	112.00
*B_2	112.00
*$\underline{B}_5\infty F_3$	110.00
*$F_5\infty\underline{B}_2$	106.00
*$\underline{B}_5F_3$	106.00
*B_1	106.00
*$F_5\underline{B}_2$	104.00
Id(Bk)	70.00

Table 5.16: Excerpt of final ranking values for North Estonian after simulation with Learner C.

There is also a great deal more space between Id-σ_1(Bk) and Id(Bk), allowing for more-categorical relationships between the constraints of interest. For example, when given the ungrammatical test input /y...æ/, this grammar selects the intended output [y...ɑ] in 100% of test evaluations, as shown in Tableau (116) (compared with only 65% correct for Learner B where Id-σ_1(Bk) and Id(Bk) were separated by $d = 20$, as shown in Table 5.10).

(116) Sample evaluation of test input /y...æ/ in the North Estonian grammar acquired by Learner C (with ranking value shown for each constraint). The grammar successfully selects the candidate without a marked F_3 vowel in the second syllable.

/y...æ/	Id-σ_1(Bk): 124	*F_3: 112	*B_1: 106	Id(Bk): 70
a. y...æ		∗∗!		
☞ b. y...ɑ		∗		∗
c. u...æ	∗!	∗		∗
d. u...ɑ	∗!			∗∗

Finnish: Table 5.17 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. Once again, both faithfulness constraints have risen to the top and the final grammar does not meet the crucial rankings proposed in Section 3.2.3. The target rankings require:

$*\mathrm{B}_2 \gg \textsc{Id}\text{-}\sigma_1(\mathrm{Bk}) \gg \textsc{Id}(\mathrm{Bk})$

$*\mathrm{F}_3\underline{\mathrm{B}}_5,\ *\mathrm{F}_3{\infty}\underline{\mathrm{B}}_5,\ *\underline{\mathrm{B}}_5\mathrm{F}_3,\ *\underline{\mathrm{B}}_5{\infty}\mathrm{F}_3 \gg \textsc{Id}(\mathrm{Bk})$

Rather, the final grammar produced by Learner C on Finnish inputs is more or less fully faithful.

Constraint	Final ranking value
$\textsc{Id}\text{-}\sigma_1(\mathrm{Bk})$	116
$\textsc{Id}(\mathrm{Bk})$	112
$*\mathrm{F}_3{\infty}\underline{\mathrm{B}}_5$	110
$*\underline{\mathrm{B}}_5{\infty}\mathrm{F}_3$	110
$*\underline{\mathrm{B}}_5\mathrm{F}_3$	108
$*\mathrm{B}_2$	108
$*\mathrm{F}_3\underline{\mathrm{B}}_5$	104

Table 5.17: Excerpt of final ranking values for Finnish after simulation with Learner C.

North Seto: Table 5.18 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. Similar to the Finnish results, both faithfulness constraints have risen to the top. As for Learner B, though the markedness constraints are correctly ordered relative to each other, the relative positions of the faithfulness vs the markedness constraints are not correct with respect to the crucial rankings proposed in Section 3.2.4. The target rankings require:

$\textsc{Id}\text{-}\sigma_1(\mathrm{Bk}) \gg *\mathrm{B}_1 \gg \textsc{Id}(\mathrm{Bk})$

$*\mathrm{F}_4\underline{\mathrm{B}}_5,\ *\mathrm{F}_4{\infty}\underline{\mathrm{B}}_5,\ *\underline{\mathrm{B}}_5\mathrm{F}_4,\ *\underline{\mathrm{B}}_5{\infty}\mathrm{F}_4 \gg \textsc{Id}(\mathrm{Bk})$

Constraint	Final ranking value
$\textsc{Id}\text{-}\sigma_1(\mathrm{Bk})$	118.00
$\textsc{Id}(\mathrm{Bk})$	114.22
$*\mathrm{F}_4{\infty}\underline{\mathrm{B}}_5$	112.00
$*\underline{\mathrm{B}}_5{\infty}\mathrm{F}_4$	112.00
$*\mathrm{F}_4\underline{\mathrm{B}}_5$	110.00
$*\underline{\mathrm{B}}_5\mathrm{F}_4$	110.00
$*\mathrm{B}_1$	108.02

Table 5.18: Excerpt of final ranking values for North Seto after simulation with Learner C.

As noted earlier in this section (with reference to Table 5.15), the average rate of correct results is higher than for the grammar acquired by Learner B, even though the final ranking values show a constraint ordering that appears to be fully faithful. This difference is due to the spacing between constraints– in particular, $\textsc{Id}(\mathrm{Bk})$ vs $*\mathrm{B}_1$ and the crucial VH constraints ($*\mathrm{F}_4{\infty}\underline{\mathrm{B}}_5$, $*\underline{\mathrm{B}}_5{\infty}\mathrm{F}_4$, $*\mathrm{F}_4\underline{\mathrm{B}}_5$, and $*\underline{\mathrm{B}}_5\mathrm{F}_4$). In the final grammar acquired by Learner C, all of the constraints are much closer together and therefore the stochastic evaluation is more likely to result in Id(Bk) occasionally swapping places with one or more of the markedness constraints, generating outputs that obey markedness (vowel harmony and/or positional restrictions) rather than faithfulness pressures.

The next section investigates the combination of the two $F_{spec} >> F_{gen}$ biases as tested thus far, and overall evaluation of all possible combinations of settings is addressed in Section 5.5.

5.2.3 Simulation results - *a priori* and Favour Specificity

The results presented in Sections 5.2.1 and 5.2.2 show that the Favour Specificity bias is more useful than the *a priori* bias for North Estonian. It facilitates enough space between the faithfulness constraints for $*F_3$ and $*B_2$ to settle in between, in order to ensure that the vowels in those sets are restricted in non-initial syllables.

With respect to North Seto, in both sets of results $*B_1$ is too low to be active, but were it to have risen higher it would similarly need to be sandwiched in between the two faithfulness constraints. Similar to North Estonian, the *a priori* bias on its own does not result in quite enough space. However, the Favour Specificity on its own seems to result in even less.

If combined, the Favour Specificity bias and the *a priori* bias have the potential to both (a) facilitate growth in the amount of space between the faithfulness constraints and (b) set a lower bound on the size of that space in order to avoid Id(Bk) encroaching on Id-σ_1(Bk) in case of errors in non-initial syllables.

In this section I briefly present results from a learner with both of these biases applied. I simulate acquisition of the three sample languages using Learner D, defined with the settings in Table 5.19.

Learner D:	**Parameter**	**Setting**
	All basic parameters	Default
	A priori bias ($F_{spec} >> F_{gen}$)	$d = 20$
	Favour Specificity bias ($F_{spec} >> F_{gen}$)	Active

Table 5.19: Parameter settings for Learner D.

With both of these biases applied, learning simulations for all three sample languages still fail to acquire the target grammars, producing similar results to those from Learners B and C. Test results are summarized in Table 5.20; results and final rankings for each individual language follow.

Language	**Average rate of correct outputs (%)**
North Estonian	90.00%
Finnish	33.67%
North Seto	63.00%

Table 5.20: Summary of results from simulations with Learner D.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.21.

Finnish: Table 5.22 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data.

Constraint	Final ranking value
Id-σ_1(Bk)	128
$*\mathrm{F}_5\infty\underline{\mathrm{B}}_2$	116
$*\mathrm{B}_2$	116
$*\mathrm{F}_3$	112
$*\mathrm{F}_5\underline{\mathrm{B}}_2$	110
$*\underline{\mathrm{B}}_5\infty\mathrm{F}_3$	108
$*\mathrm{B}_1$	106
$*\underline{\mathrm{B}}_5\mathrm{F}_3$	104
Id(Bk)	80

Table 5.21: Excerpt of final ranking values for North Estonian after simulation with Learner D.

Constraint	Final ranking value
Id-σ_1(Bk)	134.01
Id(Bk)	114.01
$*\mathrm{B}_2$	112.00
$*\mathrm{F}_3\infty\underline{\mathrm{B}}_5$	110.00
$*\underline{\mathrm{B}}_5\mathrm{F}_3$	108.00
$*\underline{\mathrm{B}}_5\infty\mathrm{F}_3$	108.00
$*\mathrm{F}_3\underline{\mathrm{B}}_5$	106.00

Table 5.22: Excerpt of final ranking values for Finnish after simulation with Learner D.

North Seto: Table 5.23 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data.

Although none of these results are meaningfully different from those of Learners B or C, I will be using both together in subsequent simulations, in order to ensure both a minimum distance between faithfulness constraints, and the ability to expand that distance where motivated by first-syllable errors.

5.2.4 Discussion

The application of the *a priori* bias enables the learner to produce grammars in which specific faithfulness constraints are ranked higher than their general counterparts. In the context of Finnic languages, such a bias facilitates the kind of first-syllable privilege that languages in this typology require– whether to ensure that neutralization only occurs in non-initial syllables (as for North Estonian) or to allow for harmony to be driven by the first syllable (as for Finnish and North Seto).

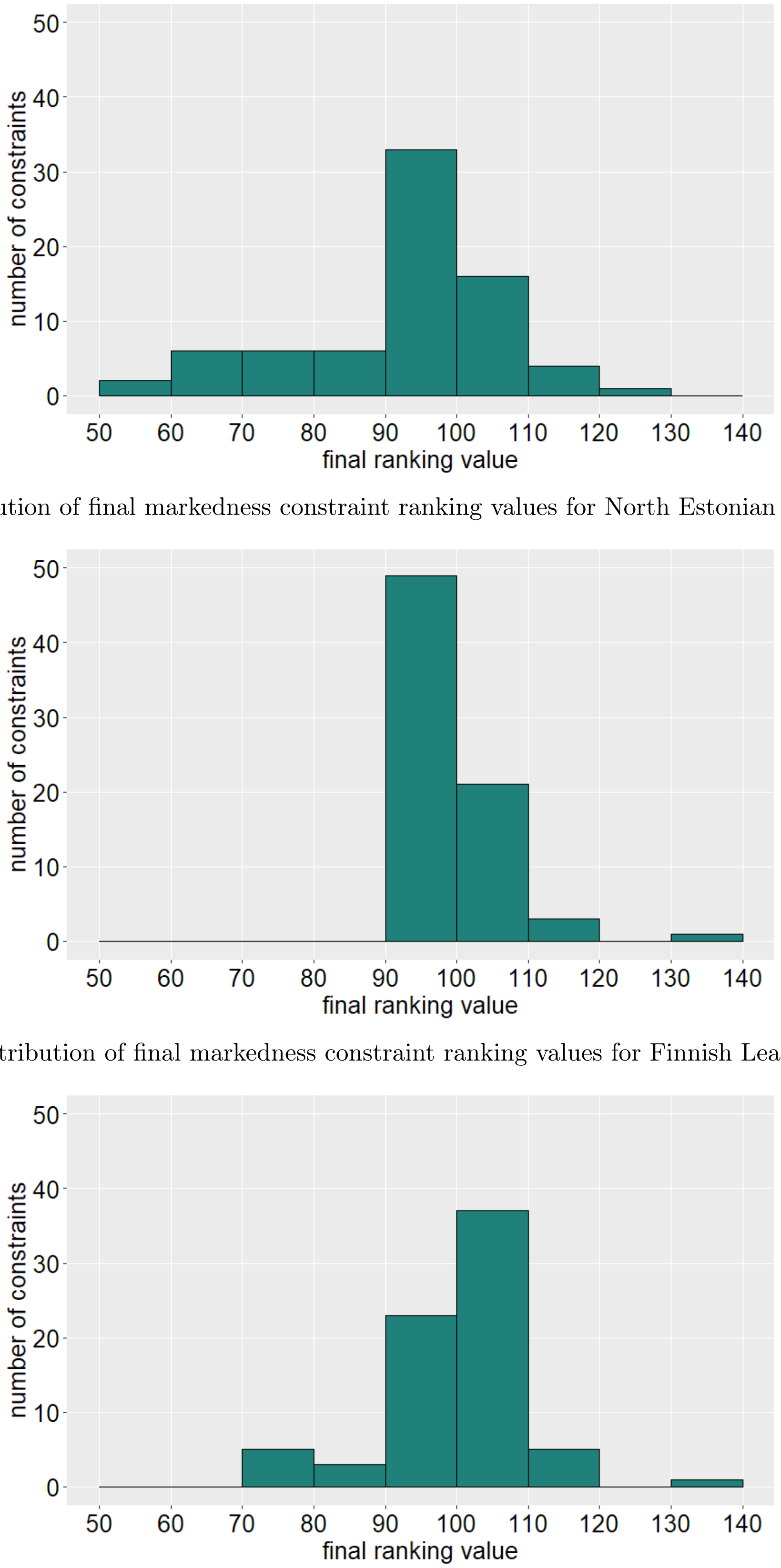
The Favour Specificity bias has a similar effect, but is more flexible, allowing specific faithfulness constraints to rise independently of their general counterparts. However, without a minimum required distance between the specific and general faithfulness constraints, there are situations in which this results in a more crowded sequence of final ranking values (for example, North Seto Learner C). This means a more variable final grammar, in a learning context where variability is not a desired characteristic of the target grammar.

Constraint	**Final ranking value**
Id-σ_1(Bk)	136.06
Id(Bk)	116.06
*$\text{F}_4\infty\underline{\text{B}}_5$	116.00
*$\underline{\text{B}}_5\infty\text{F}_4$	116.00
*$\underline{\text{B}}_5\text{F}_4$	114.00
*$\text{F}_4\underline{\text{B}}_5$	112.00
*B_1	102.00

Table 5.23: Excerpt of final ranking values for North Seto after simulation with Learner D.

Even with the degree of success shown by the North Estonian Learners B and C, it is clear that neither of these biases (or even both together) is enough for successful learning of the sample languages. A particular obstacle that recurs consistently in the simulations discussed throughout Section 5.2 is that most of the markedness constraints do not shift away from their initial values to any great degree (Figure 5.4). There are a number of reasons for this, which are discussed below, but the overarching consequence is that most of the relative rankings between the various markedness constraints do not have the opportunity to become anywhere near categorical. With constantly shifting markedness pressures and steadily rising faithfulness constraints, the learner cannot determine which markedness constraints to credit with any successful outputs and is only able to start selecting the intended winners as optimal once the faithfulness constraints have surpassed the chaos of the markedness constraints. After that point, since the learners receive only positive evidence, the faithfulness constraints continue to get credit for any winners, rendering the markedness constraints effectively useless.

Figure 5.4: Distribution of final markedness constraint ranking values for Learners D (all markedness constraints were initialized at 100).



(a) Distribution of final markedness constraint ranking values for North Estonian Learner D.

(b) Distribution of final markedness constraint ranking values for Finnish Learner D.

(c) Distribution of final markedness constraint ranking values for North Seto Learner D.

Due to the nearsightedness of the GLA and the use of positive evidence only, markedness constraints that are never violated by the learning data (e.g., $*\underline{B}_5{\infty}F_3$ in Finnish) are highly unlikely to ever be violated by a generated output; the only way this would happen is due to evaluation noise. Thus they have negligible opportunity to be promoted as a result of such an error. Ideally, to compensate, the other markedness constraints would instead *fall* from their initial values. However, that turns out to be unlikely as well. For example, the symmetrical properties of $*F_5$ and $*B_5$ result in these two constraints staying quite steady relative to each other, and also fairly close to their initial value, as errors that promote one demote the other and vice versa;[42] see (117). The rest of the context-free scale referring constraints $*F_m$ and $*B_n$, while not perfectly symmetric, are antagonistic enough to result in approximately similar behaviour.

(117) Violation profile for a sample learning error with North Seto input.

/u...o/	$*B_5$	$*F_5$	...	Id-σ_1(Bk)	Id(Bk)
✓ a. u...o	**		...		
☞ b. u...ø	*L	*W	...		*W

As for the no-disagreement constraints, there is a slightly different issue at play. The vowel harmony constraints that are often violated and should be inactive in the target grammar do get demoted as errors are made in which they prefer the intended losers. For example, consider $*F_5\underline{B}_5$ in Finnish. Since /i/ and /e/, both in set F_5, are transparent in Finnish, such a constraint is violated by many of the learning inputs (e.g., /u...i...o/) and is therefore demoted when it contributes to an error. For example, (118a) and (118b) show examples of the kinds of errors that demote $*F_5\underline{B}_5$ in Finnish, as desired.

(118) ERC excerpts for errors demoting $*F_5\underline{B}_5$ in Finnish.

input	*winner ∼ loser*	$*F_5$	$*B_5$	$*F_5\underline{B}_5$
a. /e...ɑ/	e...ɑ ∼ e...æ	W	L	L
b. /u...i...o/	u...i...o ∼ u...i...ø	W	L	L

But, given the complexity of this constraint set, there are other forces re-promoting these types of constraints. For example, consider again $*F_5\underline{B}_5$ in Finnish. As mentioned above, it does get demoted when it contributes to an error. On the other hand, it is also often promoted as a side effect of updates related to other errors, to such an extent that much of the downward movement is cancelled out. (119a) and (119b) show examples of such errors, assuming faithfulness to the first syllable.

(119) ERC excerpts for errors “accidentally” promoting $*F_5\underline{B}_5$ in Finnish.

input	*winner ∼ loser*	$*F_5$	$*B_5$	$*F_3\underline{B}_5$	$*F_5\underline{B}_5$
a. /i...ø/	i...ø ∼ i...o	L	W	e	W
b. /æ...ø/	æ...ø ∼ æ...o	L	W	W	W

In (119a), both candidates are grammatical but only the intended winner is faithful to the input. Therefore when the learner selects the loser as optimal in order to avoid violating the currently-highly-ranked $*F_5$, the update promotes $*F_5\underline{B}_5$ even though it had nothing to do with the selection of the winner and is in fact very reasonable to violate in Finnish.

[42] In this constraint set, the only possible repair for markedness violations is to change the backness of a vowel. Hence, avoiding a violation of (e.g.) $*F_5$ means incurring a violation of Id(Bk) and therefore also of $*B_5$.

In (119b), the learner again selects the loser in order to avoid violating $*\mathrm{F}_5$. In this case, the resulting promotion of $*\mathrm{F}_5\underline{\mathrm{B}}_3$ is desired. But due to the no-disagreement constraints being built up from the nested stringency sets, the loser's violation (and resulting promotion) of $*\mathrm{F}_5\underline{\mathrm{B}}_3$ also necessarily means promotion of superset-referring $*\mathrm{F}_5\underline{\mathrm{B}}_5$.

Whether considering the context-free markedness constraints or the no-disagreement constraints, either way we run the risk of producing a strictly faithful grammar (which accounts for all of the learning data but no potential unfaithful test data) if the general faithfulness constraint is permitted to rise above the markedness constraints as their values oscillate. The need for more space between the (ideally) higher-ranked markedness constraints and the lower ones is hindered by their oscillation, but can be facilitated by defining a learner with asymmetry between its promotion vs demotion amounts. The idea is for ERCs such as those in (118) to be more influential than those in (119). This adaptation – a modified update rule – is presented in Section 5.3.

5.3 Promotion rate

GLA-type learners make adjustments to the ranking values after each error made by the current hypothesized grammar. While all variations on this theme agree that constraint *demotion* is necessary to the learning process, arguments have been made both for (e.g., Boersma, 1997, 1998; Magri, 2012) and against (e.g., Tesar & Smolensky, 1998) the idea of permitting constraint *promotion* as well. I subscribe to Magri's (2012) claim that some promotion must be required in order to allow for re-ranking of faithfulness constraints which, in a learning environment that assumes faithful underlying forms for licit inputs, never prefer losers.

As Magri (2012) does, I determine the promotion amount for winner-preferring constraints as a fraction of the current plasticity. E.g.,

$$\textit{promotion amount} = (\textit{promotion rate}) \times \textit{plasticity} \tag{5.1}$$

where promotion rate is a fraction determined as a function of the number of winner-preferring and/or loser-preferring constraints at that update.

5.3.1 Rationale

At the low end, a promotion rate of 0 means that initially low faithfulness constraints would remain stuck at their starting values; they need some way to rise to allow for adjustments to rankings as new learning inputs are encountered. At the high end, a promotion rate of 1 (or more) means that every winner-preferring constraint is given full credit for preference of the winner. However, we should consider avoiding full-fledged promotion of constraints in the case of an ERC that contains two or more constraints that prefer the intended winner, in order to avoid overpromoting when it is not clear which of those constraints should be credited with preference of the winner (the Credit Problem; Dresher, 1999). Between these two extremes, Magri (2012) argues that different ranges of promotion rates have been shown to result in efficient convergence, inefficient convergence, or non-convergence of a GLA-type learner; see Figure 5.5. The value that delineates these ranges is shown by Magri to be the ratio l/w, where w = number of winner-preferring constraints[43] and l = number of loser-preferring constraints (these variables will be used in calculations of promotion rate throughout this section).

[43] With reference to the Favour Specificity bias defined in Section 5.2.2, note that even when the winner-preferring Id(Bk) is not promoted, it does still count toward w, the total number of winner-preferring constraints.

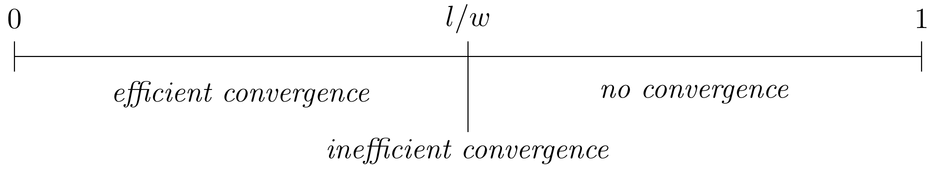


Figure 5.5: Convergence and efficiency depend on promotion rate (adapted from Magri, 2012 (66), p. 265).

The goal, then, is to select a function for promotion rate $\in (0, 1)$ and ideally $\in (0, l/w)$,[44] such that the greater the number of constraints to be promoted, the smaller the promotion amount. Various options have been tested and are detailed in Section 5.3.2.

5.3.2 Implementation

At each learning update, the numbers of winner-preferring and loser-preferring constraints are determined from the relevant ERC. The ranking values of the loser-preferrers are decreased by the plasticity amount, and those of the winner-preferrers are increased by the promotion rate as a fraction of the plasticity. The default value for promotion rate is 1 (that is, promotion amount is equal to plasticity). I also consider four different promotion rates as functions of the number of loser- and/or winner-preferring constraints; they are described below. All four of these promotion rates are $\in (0, l/w)$ (proofs in Appendix A.1.1) and therefore satisfy Magri's (2012) requirements for efficient convergence.[45]

Magri (2012) proposes a *calibrated*[46] promotion rate:

(120) Type 1: promotion rate $= l/(1 + w)$

However, this particular rate has the potential to produce fractions greater than one, meaning that constraints could be promoted by an even larger amount than they are demoted. This goes against the general goal of avoiding overzealous promotion of winner-preferring constraints (i.e., the Credit Problem) as discussed in Section 5.3.1.

I test three additional options for a tempered promotion rate, each guaranteed to produce a fraction no greater than one (proofs in Appendix A.1.1):

(121) Type 2: promotion rate $= l/(l + w)$

(122) Type 3: promotion rate $= 1/w$

(123) Type 4: promotion rate $= 1/(1 + w)$ (Magri & Kager, 2015)

Figure 5.6 shows, for $w, l \in [1, 10]$, how the promotion rates vary as calculated by each of the four promotion rate functions above (and starting with the default, constant promotion rate of 1).

[44] Figure 5.5 implies that $0 < l/w < 1$, but in fact this is only true if $l < w$. So it is not possible to make a generally applicable statement about whether l/w or 1 is the tighter upper bound. At any rate, I aim to satisfy both requirements.

[45] Except for $1/w$ (Type 3), which has a *very* slightly higher bound and is instead $\in (0, l/w]$.

[46] As per Magri (2012), a calibrated promotion rate is one that is $< l/w$.

The default promotion rate, Type 3, and Type 4 do not depend on l and therefore only have one possible outcome for each w value. Types 1 and 2 depend on both w and l; therefore each of the corresponding plots shows ten different possible outcomes for each w value. Although the range of values included for w and l are merely for illustrative purposes and do not hold any particular significance with respect to the ERCs involved in these simulations, these plots are intended to help the reader visualize the degree of consistency resulting from each choice of promotion rate type.

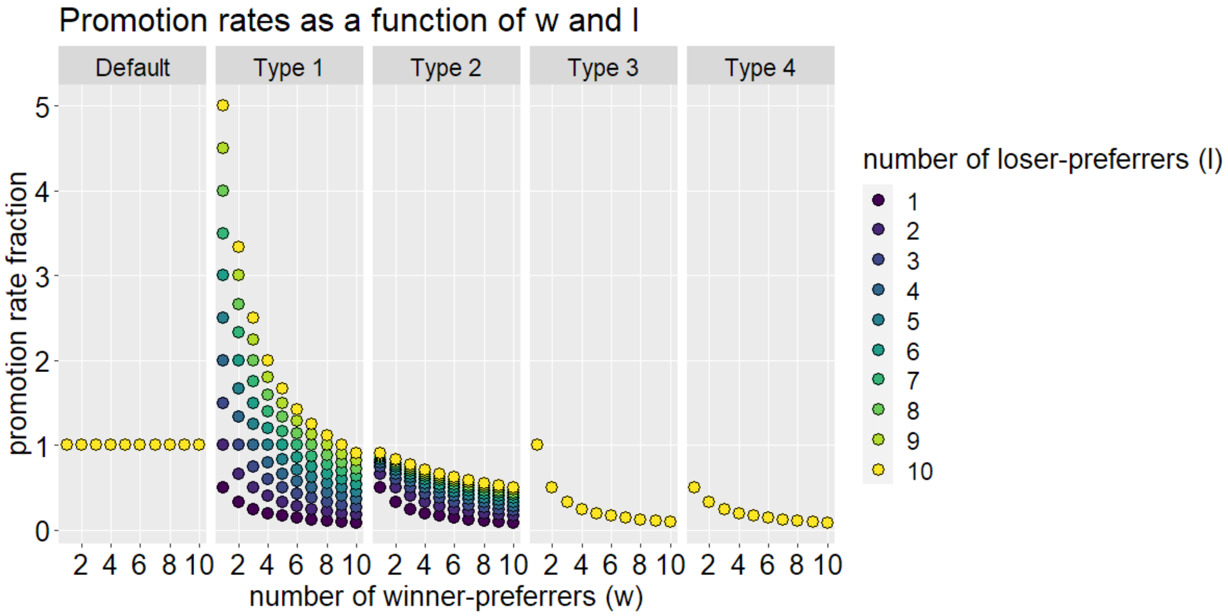


Figure 5.6: Varying promotion rate through five different types (default and 1 through 4), as a function of w and l.

5.3.3 Simulation results - promotion rate

To demonstrate the effect of a tempered promotion rate, I simulate acquisition of the three sample languages using Learner E, defined with the settings in Table 5.24. The choice of $1/w$ for the promotion rate is for illustrative purposes and is strictly based on the simplicity of the expression rather than any other factor. Results using learners with the other types are summarized in Section 5.5.

Learner E:	**Parameter**	**Setting**
	All basic parameters	Default
	Promotion rate	$1/w$

Table 5.24: Parameter settings for Learner E.

With the modified promotion rate applied – even without the initial help of the specific-over-general faithfulness bias – the results for the vowel harmony languages (and especially Finnish) are an improvement on those from earlier learners. Results are summarized in Table 5.25. Detailed commentary on each language follows, and a bigger-picture summary of the effects of tempered promotion rate is discussed in Section 5.3.5.

However, inspection of the final ranking values reveals some interesting details not conveyed in these

Language	Average rate of correct outputs (%)
North Estonian	62.64%
Finnish	90.98%
North Seto	73.27%

Table 5.25: Summary of results from simulations with Learner E.

summary results. These details are discussed below, language by language.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.26.

Constraint	Final ranking value
*$\underline{B}_5F_3$	100.15
*$\underline{B}_5\infty F_3$	100.15
*$F_5\underline{B}_2$	100.00
*$F_5\infty\underline{B}_2$	100.00
*B_1	100.00
Id(Bk)	75.55
Id-σ_1(Bk)	53.30
*B_2	48.88
*F_3	-17.03

Table 5.26: Excerpt of final ranking values for North Estonian after simulation with Learner E.

The final grammar shows that this learner, due in part to only having access to positive evidence, has "misunderstood" North Estonian to be a vowel harmony language. With the modified promotion rate ensuring that demotions are larger than promotions, the restricted vowels in non-initial syllables are accounted for by no-disagreement constraints (*$F_5\underline{B}_2$, *$F_5\infty\underline{B}_2$, *$\underline{B}_5F_3$, *$\underline{B}_5\infty F_3$) rather than context-free constraints (*F_3, *B_2). This happens because the no-disagreement constraints are never violated by the learning data (and therefore almost never move), while the relevant context-free constraints are violated quite often, specifically by vowels in the first syllable (and therefore move downward anytime such a first-syllable vowel is encountered). The ERC matrix in (124) shows that when the input happens to contain two relatively unmarked back vowels (/u...ɑ/), both *F_3 and *$\underline{B}_5F_3$ are winner-preferring and therefore promoted, but when the input sequence contains a marked front vowel (/y...æ/), *F_3 is demoted while *$\underline{B}_5F_3$ does not move.

(124) ERC matrix comparing inputs with marked vs unmarked vowels in the initial syllable.

input	candidates	Id-σ_1(Bk)	Id(Bk)	*F_3	*$\underline{B}_5F_3$	*B_5
/u...ɑ/	u...ɑ ∼ u...æ	e	W	W	W	L
/y...ɑ/	y...ɑ ∼ u...ɑ	W	W	L	e	W

*B_1 does successfully implement an inventory gap as required, but the loose vowel harmony attested by this grammar is not driven by faithfulness to the first syllable; rather, it is driven by general faithfulness over all segments in the word. Even if the first-syllable faithfulness was ranked higher,

interpreting North Estonian as a vowel harmony language rather than one with broader positional restrictions would mean that vowels in sets F_3 and B_2 are banned only when following a vowel of the opposite backness. Hence patterns such as [y...æ] are deemed acceptable even though such sequences are not attested in North Estonian; see Tableau (125).

(125) Sample evaluation of test input /y...æ/ in the North Estonian grammar acquired by Learner E. The grammar selects the candidate in which both vowels are front (i.e., the harmonic candidate), even though it has a vowel in set F_3 in a non-initial syllable.

/y...æ/	$*\underline{B}_5F_3$	$*\underline{B}_5\infty F_3$	Id(Bk)	Id-σ_1(Bk)	$*F_3$
☞ a. y...æ					∗∗
✓ b. y...ɑ			∗!		∗
c. u...æ	∗!	∗	∗	∗	∗
d. u...ɑ			∗!∗	∗	

Finnish: Table 5.27 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. As for several of the previous learners, the key markedness constraints are at the top of the rankings, but in this case both faithfulness constraints are significantly lower than the inventory gap $*B_2$ and the F_3/B_5 vowel harmony constraints. This grammar does almost exactly what is needed for Finnish: it completely bans vowels in set B_2 (/ɯ, ɤ/) and ensures the required harmony and transparency patterns among the remaining vowels. However, similar to the North Estonian result, it does so in a way that is not governed by context-specific (initial syllable) faithfulness. Therefore winners are always harmonic, but whether they are back or front is determined by either general faithfulness (the harmonic candidate with fewer faithfulness violations is preferred over a different harmonic candidate with more faithfulness violations[47]) or, if faithfulness violations are equal, then by other lower-ranked markedness constraints (whether context-free or no-disagreement) that should not ideally be active. Tableaux (126) and (127) show examples of each of these cases.

Constraint	**Final ranking value**
$*B_2$	101.30
$*F_3\underline{B}_5$	100.00
$*F_3\infty\underline{B}_5$	100.00
$*\underline{B}_5F_3$	100.00
$*\underline{B}_5\infty F_3$	100.00
(several other unviolated VH constraints)	100.00
Id(Bk)	64.90
$*F_1$	55.04
Id-σ_1(Bk)	42.66

Table 5.27: Excerpt of final ranking values for Finnish after simulation with Learner E.

[47] This non-attested pattern is known as Majority Rule harmony (Baković, 2000; Lombardi, 1999).

(126) Sample evaluation of test input /ø...o/ in the Finnish grammar acquired by Learner E. The grammar selects the candidate with fewer violations of $*F_1$ even though it is not faithful to the first syllable, nor should such a constraint be active in this language.

/ø...o/	$*F_3\underline{B}_5$	$*F_3\infty\underline{B}_5$	$*\underline{B}_5F_3$	$*\underline{B}_5\infty F_3$	Id(Bk)	$*F_1$	Id-σ_1(Bk)
a. ø...o	*!	*				*	
✓ b. ø...ø					*	*!*	
c. o...ø			*!	*	**	*	*
☞ d. o...o					*		*

(127) Sample evaluation of test input /ø...ɑ...u/ in the Finnish grammar acquired by Learner E. The grammar selects the candidate with fewer overall violations of Id(Bk) even though it is not faithful to the first syllable. Disharmonic candidates are omitted from this tableau for the sake of simplicity, since they are ruled out immediately by the highest-ranked no-disagreement constraints.

/ø...ɑ...u/	$*F_3\underline{B}_5$	$*F_3\infty\underline{B}_5$	$*\underline{B}_5F_3$	$*\underline{B}_5\infty F_3$	Id(Bk)	$*F_1$	Id-σ_1(Bk)
✓ a. ø...æ...y					**!	*	
☞ b. o...ɑ...u					*		*

North Seto: Table 5.28 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. The key vowel harmony constraints are successfully at the top of the rankings, ensuring the required harmony and transparency patterns. The positional restriction $*B_1$ is also below Id-σ_1(Bk) as it should be; however, because the general faithfulness constraint is above the specific one, two problems arise. First, winners are always harmonic but their backness is determined by general faithfulness before first-syllable faithfulness; see Tableau (128). Second, $*B_1$ is too far down to have the opportunity to be active, so there is in effect no restriction on non-initial syllables; see Tableau (129).

Constraint	**Final ranking value**
$*\underline{B}_5F_4$	100.29
$*\underline{B}_5\infty F_4$	100.29
$*F_4\underline{B}_5$	100.00
$*F_4\infty\underline{B}_5$	100.00
Id(Bk)	70.30
Id-σ_1(Bk)	50.99
$*B_1$	43.98

Table 5.28: Excerpt of final ranking values for North Seto after simulation with Learner E.

(128) Sample evaluation of test input /æ...ɯ...ɤ/ in the North Seto grammar acquired by Learner E. The grammar selects the candidate with fewest overall violations of Id(Bk) even though it both contains a non-initial [ɯ] and is not faithful to the first syllable. Disharmonic candidates are omitted from this tableau for the sake of simplicity, since they are ruled out immediately by the highest-ranked no-disagreement constraints.

/æ...ɯ...ɤ/	$*F_4\underline{B}_5$	$*F_4\infty\underline{B}_5$	$*\underline{B}_5F_4$	$*\underline{B}_5\infty F_4$	Id(Bk)	Id-σ_1(Bk)	$*B_1$
✓ a. æ...i...e					$**!$		
☞ b. ɑ...ɯ...ɤ					$*$	$*$	$*$
c. ɑ...i...ɤ					$**!$	$*$	

(129) Sample evaluation of test input /ɯ...ɯ/ in the North Seto grammar acquired by Learner E. Since the input is already harmonic, the grammar selects the candidate with the fewest faithfulness violations, even though [ɯ] is restricted and should not appear in the second syllable.

/ɯ...ɯ/	$*F_4\underline{B}_5$	$*F_4\infty\underline{B}_5$	$*\underline{B}_5F_4$	$*\underline{B}_5\infty F_4$	Id(Bk)	Id-σ_1(Bk)	$*B_1$
☞ a. ɯ...ɯ							$**$
✓ b. ɯ...i					$*!$		$*$
c. i...ɯ					$*!$	$*$	$*$
d. i...i					$*!*$	$*$	

5.3.4 Simulation results - $\mathbf{F_{spec} \gg F_{gen}}$ and promotion rate

To demonstrate the combined effects of the *a priori* bias, Favour Specificity bias, and tempered promotion rate, I simulate acquisition of the three sample languages using Learner F, defined with the settings in Table 5.29. As before, these specific settings are used for illustrative purposes; overall evaluation is presented in Section 5.5.

Learner F:	**Parameter**	**Setting**
	All basic parameters	Default
	A priori bias ($F_{spec} \gg F_{gen}$)	$d = 20$
	Favour Specificity bias ($F_{spec} \gg F_{gen}$)	Active
	Promotion rate	$1/w$

Table 5.29: Parameter settings for Learner F.

With all three of these modifications implemented, the results show improvement yet again compared to those from prior learners. Results are summarized in Table 5.30; results and final rankings for each individual language are discussed in more detail below.

Language	**Average rate of correct outputs (%)**
North Estonian	88.53%
Finnish	100.00%
North Seto	99.97%

Table 5.30: Summary of results from simulations with Learner F.

At this point, the grammar acquired by the Finnish learner is achieving 100% success on tests, the

North Seto grammar nearly so, and the North Estonian grammar is performing quite well but with some room for improvement. Final rankings for this learner are presented and analyzed below.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.31. For comparison, recall the crucial rankings from Section 3.2.2:

$*B_1$ >> Id-σ_1(Bk) >> $*F_3$, $*B_2$ >> Id(Bk)

The final rankings of Learner F improve those of Learner D (which had the same settings minus the tempered promotion rate) in that $*B_1$ is at the top, implementing an inventory gap. They also improve those of Learner E in that Id-σ_1(Bk) outranks Id(Bk), ensuring faithfulness to the vowel in the first syllable. However, although $*B_2$ is sandwiched between the two faithfulness constraints as required, the North-Estonian-as-a-vowel-harmony-language problem still persists with highly ranked $*F_5\underline{B}_2$, $*F_5\infty\underline{B}_2$, $*\underline{B}_5F_3$, and $*\underline{B}_5\infty F_3$, while $*F_3$ is below general faithfulness and therefore inactive (though Id(Bk) and $*F_3$ are close enough in ranking value to swap places some of the time due to evaluation noise). Tableaux (130) and (131) show how vowels in set B_2 are appropriately restricted in non-initial syllables but those in set F_3 may not be.

Constraint	**Final ranking value**
$*F_5\underline{B}_2$	100.00
$*F_5\infty\underline{B}_2$	100.00
$*\underline{B}_5F_3$	100.00
$*\underline{B}_5\infty F_3$	100.00
$*B_1$	100.00
Id-σ_1(Bk)	70.98
$*B_2$	60.80
Id(Bk)	39.15
$*F_3$	37.63

Table 5.31: Excerpt of final ranking values for North Estonian after simulation with Learner F.

(130) Sample evaluation of test input /ɑ...ɤ/ in the North Estonian grammar acquired by Learner F. The grammar successfully selects the candidate that avoids [ɤ] in the second syllable.

/ɑ...ɤ/	$*F_5\underline{B}_2$	$*F_5\infty\underline{B}_2$	$*\underline{B}_5F_3$	$*\underline{B}_5\infty F_3$	Id-σ_1(Bk)	$*B_2$	Id(Bk)	$*F_3$
a. ɑ...ɤ						*!		
☞ b. ɑ...e							*	
c. æ...ɤ	*!	*			*	*	*	
d. æ...e					*!		**	

(131) Sample evaluation of test input /y...ø/ in the North Estonian grammar acquired by Learner F. The grammar selects the candidate in which both vowels are front, even though it has a vowel in set F$_3$ in a non-initial syllable.

/y...ø/	*F$_5\underline{\text{B}}_2$	*F$_5\infty\underline{\text{B}}_2$	*$\underline{\text{B}}_5$F$_3$	*$\underline{\text{B}}_5\infty$F$_3$	Id-σ_1(Bk)	*B$_2$	Id(Bk)	*F$_3$
☞ a. y...ø								∗∗
✓ b. y...o							∗!	∗
c. u...ø			∗!	∗	∗		∗	∗
d. u...o					∗!		∗∗	

Finnish: Table 5.32 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. This grammar meets all the requirements for a target Finnish grammar, shown in full in Figure 5.2 and with these crucial highest rankings:

*B$_2$ >> Id-σ_1(Bk) >> Id(Bk)

*F$_3\underline{\text{B}}_5$, *F$_3\infty\underline{\text{B}}_5$, *$\underline{\text{B}}_5F_3$, *$\underline{\text{B}}_5\infty$F$_3$ >> Id(Bk)

The ranking values are far enough apart in value to behave effectively categorically, as evidenced by the 100% rate of correct outputs during testing.

Constraint	Final ranking value
*B$_2$	100.80
*F$_3\underline{\text{B}}_5$	100.00
*F$_3\infty\underline{\text{B}}_5$	100.00
*$\underline{\text{B}}_5$F$_3$	100.00
*$\underline{\text{B}}_5\infty$F$_3$	100.00
Id-σ_1(Bk)	68.96
Id(Bk)	43.48

Table 5.32: Excerpt of final ranking values for Finnish after simulation with Learner F.

North Seto: Table 5.33 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. This grammar meets all the requirements for a target North Seto grammar, namely:

Id-σ_1(Bk) >> *B$_1$ >> Id(Bk)

*F$_4\underline{\text{B}}_5$, *F$_4\infty\underline{\text{B}}_5$, *$\underline{\text{B}}_5F_4$, *$\underline{\text{B}}_5\infty$F$_4$ >> Id(Bk)

However, some pairs of ranking values are just close enough together that evaluation noise results in the odd swapped ranking. For example, *B$_1$ is about 8 below Id-σ_1(Bk) and thus there are a small number of instances where inputs with a (perfectly reasonable) first-syllable /ɯ/ surface with an [i] instead, due to a temporary ranking of *B$_1$ >> Id-σ_1(Bk) at evaluation time. See Tableau (132).

Constraint	Final ranking value
$*F_4\underline{B}_5$	100.00
$*F_4\infty\underline{B}_5$	100.00
$*\underline{B}_5F_4$	100.00
$*\underline{B}_5\infty F_4$	100.00
ID-σ_1(Bk)	70.29
$*B_1$	62.38
ID(Bk)	44.22

Table 5.33: Excerpt of final ranking values for North Seto after simulation with Learner F.

(132) Sample evaluation of input /ɯ...o/ in the North Seto grammar acquired by Learner F, when $*B_1$ and ID-σ_1(Bk) are stochastically swapped at evaluation. The grammar incorrectly selects the candidate without the marked vowel /ɯ/.

/ɯ...o/	VH constraints	$*B_1$	ID-σ_1(Bk)	$*\underline{B}_1F_5$	ID(Bk)
✓ a. ɯ...o		∗!			
☞ b. i...o			∗		∗
c. i...ø			∗		∗∗!

5.3.5 Discussion

Modifying the update rule by adjusting the learning algorithm's promotion rate serves to create space between markedness constraints by ensuring that any oscillating constraint values oscillate on an overall downward trajectory (at least in this case). For a constraint set such as this one, where the use of stringency sets entails the existence of pairs of antagonistic – or near-antagonistic – constraints, this facilitates the differentiation of constraints whose values stay relatively stable due to never being violated by the positive-evidence learning inputs (such as $*\underline{B}_5F_3$ in Finnish) versus constraints whose values stay relatively stable as a result of oscillation due to repeated violation (such as $*F_5$ and $*B_5$ in any of the sample languages).

The effects of the tempered promotion rate can be understood via comparison of either Learner A with Learner E, or Learner D with Learner F, for any language. See Table 5.34 for a summary of references to earlier results tables, along with overall success rates for each learner. Comparing the final rankings of those learners reveals that with the modified promotion rate, markedness constraints maintain enough relative height that they can be active in accounting for the patterns attested in the input data, rather than leaving that work entirely to the faithfulness constraints. This is facilitated by virtue of the markedness constraints' relative rankings becoming clearly articulated enough to reduce the error rate, before faithfulness constraints rise so far as to overcome the markedness constraints.

For the vowel harmony languages, Finnish and North Seto, Learner F achieves near-perfect results. However, for North Estonian with its positional restrictions, Learner F acquires a grammar that accounts for the systematic absence of certain vowels from non-initial syllables via no-disagreement constraints rather than context-free markedness constraints. This occurs because these particular no-disagreement constraints are never violated, while the context-free constraints are part of the antagonistic collection of constraints that oscillate downward due to their repeated violation by

Learner	Settings	Results		
		North Estonian	**Finnish**	**North Seto**
A	Default	24.55% Table 5.4	26.83% Table 5.5	29.84% Table 5.6
E	Default + Promotion rate	62.64% Table 5.26	90.98% Table 5.27	73.27% Table 5.28
D	Default + *A priori* + Favour Specificity	90.00% Table 5.21	33.67% Table 5.22	63.00% Table 5.23
F	Default + *A priori* + Favour Specificity + Promotion rate	88.53% Table 5.31	100.00% Table 5.32	99.97% Table 5.33

Table 5.34: References to earlier results tables, for the reader's convenience.

the learning data. The problem with such a result is that the no-disagreement constraints are too specific: they account perfectly well for the input data but fail to generate correct outputs when given ungrammatical test data. The vowels in non-initial syllables need to be restricted not only when they are disharmonic with the vowel in the initial syllable, but also even when they happen to harmonize with it. This is in essence a restrictiveness problem: in the same way that prioritizing specific faithfulness constraints over general ones allows for a more restrictive grammar, it is clear here that more general markedness constraints need to be prioritized over more specific ones in order to ensure better restrictiveness in terms of markedness as well.

There is a wide range of potential strategies for prioritizing generality in markedness constraints; in Section 5.4 I propose several options for implementing such a bias and present results of one novel implementation.

5.4 $\mathbf{M_{gen} \gg M_{spec}}$

My constraint set does not have any explicitly defined pairs of specific vs general markedness constraints like it does faithfulness constraints. However, it is nevertheless possible to determine the relative generality of various pairs of markedness constraints, a task that is made easier due to the fact that all of the markedness constraints refer to the same stringency scales (for example, $*B_1$ is more specific than $*B_2$ since $B_1 \subset B_2$). Using a general-over-specific markedness bias works toward the same goal as the specific-over-general faithfulness bias: learning a grammar that is as restrictive as possible.

5.4.1 Rationale

Learners E and F have difficulty learning a correct ranking for Northern Estonian because the vowel harmony (no-disagreement) constraints are able to explain the limited vowels in non-initial syllables in the learning data without also incurring violations elsewhere like the context-free segmental constraints do. Although the context-free constraints are sometimes violated (specifically, by vowels in the first syllable), in fact they are much more general and can better deal with ungrammatical inputs than the no-disagreement constraints can. The more general constraints make the grammar more restrictive, which is useful for avoiding overgeneration when encountering ungrammatical inputs.

The rationale for a general-over-specific markedness bias is to give the most general markedness constraints an opportunity to get credit for the phonotactics of the target grammar, in order to ensure maximal restrictiveness.

The preference for more general markedness constraints is not persistent. Rather, I implement it as an initial articulated hierarchy of markedness constraint values that can be freely reversed by learning data. Such a hierarchy can be determined either by a set-theoretic approach or as a function of each constraint's rate of application in a sample set of inputs, both of which options are described in Section 5.4.2.

5.4.2 Implementation

Within the scale-referring markedness constraints central to this project, there are several dimensions on which generality can be measured, informed by set theory. Using these dimensions, one can determine the relative place of various markedness constraints in the initial ranking values.

- Dimension 1: size of stringency set.
    - Dimension 1a: size of stringency set (absolute). E.g., $*B_5$ is more general than $*B_2$, because it bans the whole class of back vowels rather than just two of them.
    - Dimension 1b: size of stringency set (in context). E.g., $*\underline{B}_5F_5$ is more general than $*\underline{B}_5F_2$, because it bans back vowels followed by any of the five front vowels rather than by either of just two of them.
- Dimension 2: context-sensitivity. E.g., $*B_5$ is more general than $*\underline{B}_5F_2$ and $*\underline{B}_5\infty F_2$, because it bans all instances of back vowels, not just when they precede a front vowel from set F_2.
- Dimension 3: scope of application: E.g., $*\underline{B}_5\infty F_2$ is more general than $*\underline{B}_5F_2$, because it bans all sequences including an earlier back vowel from set B_5 and a later front vowel from set F_2, not just when they are in adjacent syllables.

Using these dimensions, it is straightforward to establish dozens of intersecting generality-based hierarchies of markedness constraints. However, it is very difficult to determine how the constraints in those separate hierarchies should interleave in order to create an overall initial distribution of markedness constraints based on generality. For example, $*B_5$ is more general than $*B_2$, and $*B_5$ is more general than $*\underline{B}_5F_2$, but the relationship between $*B_2$ and $*\underline{B}_5F_2$ is not clear.

Given the difficulty of calculating the overall distribution of markedness based on their set-theoretic relationships, I propose instead a number of different methods for the learner to determine the initial relative rankings. Sections 5.4.2.1 and 5.4.2.5 present a uniform and a random distribution, to use as reference points for the generality-based methods detailed in Sections 5.4.2.2 through 5.4.2.4. The general approach is demonstrated in Section 5.4.3 using one specific implementation, and assessment of the most effective options is undertaken in Section 5.5.

5.4.2.1 Uniform distribution function

Distribution Function 1 ($\mathscr{F}_1$): The first approach is to assume that all markedness constraints have the same starting value. This is a baseline that is not informed by relative generality at all. Within $\mathscr{F}_1$, I test three different uniform starting values of 100, 300, and 500. Note that this distribution function, with a starting value of 100, corresponds to the low-faithfulness bias presented in Section 5.1.1 and has been the default for assigning initial values to markedness constraints for all simulations discussed up to this point.

5.4.2.2 Stratified distribution function - by constraint type

Distribution Function 2 ($\mathscr{F}_2$): The second approach is to construct discrete strata of markedness constraints, based on their level of generality from the perspective of constraint type: context-free, long-distance no-disagreement constraints, and local no-disagreement constraints.

The strata are assigned by $\mathscr{F}_2$ as follows:

- Stratum 1 (the highest) contains all context-free markedness constraints. For example, $*\mathrm{F}_1$ and $*\mathrm{B}_3$.
- Stratum 2 contains all long-distance no-disagreement constraints. For example, $*\mathrm{F}_1\infty\underline{\mathrm{B}}_2$ and $*\underline{\mathrm{B}}_2\infty\mathrm{F}_5$.
- Stratum 3 (the lowest) contains all local no-disagreement constraints. For example, $*\mathrm{F}_4\underline{\mathrm{B}}_2$ and $*\underline{\mathrm{B}}_5\mathrm{F}_5$.

The initial ranking values for the strata must be specified as well. I use 140, 120, and 100 for the first, second, and third strata, respectively. These values are somewhat arbitrary but are chosen for two reasons. The first is so that the lowest markedness constraints are still at least as far above the faithfulness constraints as they are in the default low-faithfulness implementation. The second is so that the three strata are separated by a difference that is effectively equivalent to a categorical ranking (i.e., $d = 20$, as discussed in Section 5.2.1.2 for the *a priori* bias).

5.4.2.3 Stratified distribution function - by stringency set

Distribution Function 3 ($\mathscr{F}_3$): The third approach is to construct discrete strata of markedness constraints, based on their level of generality from the perspective of the cardinality of the sets referred to by each constraint. Because all markedness constraints in this project are scale-referring, it is straightforward to use the front and back vowel sets to assign strata.

In the top-down version of $\mathscr{F}_3$ (called $\mathscr{F}_{3t}$), I assign strata greedily, as follows:

- Stratum 1 (the highest) contains all markedness constraints whose largest referenced set has a cardinality of 5. For example, $*\mathrm{B}_5$, $*\underline{\mathrm{B}}_3\mathrm{F}_5$, and $*\mathrm{F}_1\infty\underline{\mathrm{B}}_5$.
- Stratum 2 contains all markedness constraints whose largest referenced set has a cardinality of 4. For example, $*\mathrm{F}_4$ and $*\underline{\mathrm{B}}_1\infty\mathrm{F}_4$ (recall that B_4 is undefined).
- Stratum 3 contains all markedness constraints whose largest referenced set has a cardinality of 3.
- Stratum 4 contains all markedness constraints whose largest referenced set has a cardinality of 2.

- Stratum 5 (the lowest) contains all of the remaining markedness constraints, which are the ones that refer only to sets with cardinality 1.

In the bottom-up version of$\mathscr{F}_3$ (called $\mathscr{F}_{3b}$), I assign strata greedily, as follows:

- Stratum 5 (the lowest) contains all markedness constraints whose smallest referenced set has a cardinality of 1. For example, $*\text{F}_1$, $*\underline{\text{B}}_1\text{F}_4$, and $*\underline{\text{B}}_3\infty\text{F}_1$.
- Stratum 4 contains all markedness constraints whose smallest referenced set has a cardinality of 2. For example, $*\text{B}_2$ and $*\underline{\text{B}}_2\text{F}_5$ (recall that F_2 is undefined).
- Stratum 3 contains all markedness constraints whose smallest referenced set has a cardinality of 3.
- Stratum 2 contains all markedness constraints whose smallest referenced set has a cardinality of 4.
- Stratum 1 (the highest) contains all of the remaining markedness constraints, which are the ones that refer only to sets with cardinality 5.

For example, $*\text{B}_4$ would be assigned to Stratum 2 (the second-highest) in either version. $*\text{F}_1\infty\underline{\text{B}}_5$, on the other hand, would be assigned to Stratum 1 in the top-down version (because the *largest* set it references contains five vowels) but Stratum 5 in the bottom-up version (because the *smallest* set it references contains one vowel).

The initial ranking values of the strata are specified as 180, 160, 140, 120, and 100 for the first through fifth strata, respectively.

5.4.2.4 Input-calibrated distribution function

Distribution Function 4 ($\mathscr{F}_4$): The fourth approach is to calculate the initial ranking values of markedness constraints via a function with a more finely distributed range. In this case, as mentioned at the beginning of section 5.4.2, it is very difficult to use the theoretical relationships between different types of constraints in order to determine the relative generality of individual pairs of constraints. Instead, I use a numerical method based on the observed application rate of each constraint within the inputs seen by the learner.

Albright and Hayes (2006) present a morphological learning problem for which they propose combined use of their Minimal Generalization Learner (to induce constraints from observed data) with the GLA (to organize those constraints based on observed data). However, they find that permitting the GLA to start with all of the induced constraints at the same height results in overly specific “junk” constraints – rather than more general ones – coming out on top in the final grammar. In order to address this challenge, they determine an initial ranking based on the generality of the induced constraints (information which is readily available from the induction phase). This approach ensures that the accidentally unviolated junk constraints start low and stay low, allowing more general constraints to do the work where possible. I use this idea as inspiration for the definition of $\mathscr{F}_4$, adapting it to a phonological rather than a morphological context, and one where there is no prior information available about generality.

Recall that in Section 5.1.1 I present the idea of the learning process taking place over four stages, each with a declining plasticity and consisting of 5 000 learning trials. For the implementation of this particular bias, I prepend a pre-learning “observation” stage, with plasticity = 0.

During the observation stage, the learner is fed randomly-sampled inputs just as it is during the learning stages. However, rather than using the current hypothesized grammar to compare the optimal output to the intended winner, the learner simply tracks the number of violations for each markedness constraint in order to calculate an average *generality measure* g_M for each markedness constraint M. Algorithm 5 describes this process.

Algorithm 5: Markedness generality observation

```
Initialize a tally for each markedness constraint.
foreach surface form observed by the learner do
    Select the violation profile for the faithful candidate.
    Add the number of violations of each markedness constraint to the tally.
Divide the tally for each markedness constraint by the total number of trials in this stage.
```

The generality (or application rate) is one of three parameters used in building the initial distribution of markedness constraints. Before the first learning stage, the generality is used to calculate the initial ranking value for each markedness constraint, modifying it from the default value of 100. The calculation of this distribution requires two additional parameters; these determine:

1. The initial ranking value corresponding to a constraint with generality 0.0. This parameter is referred to as the *y-intercept coefficient*. The tested values for the y-intercept coefficient include 0.5, 1.0, and 1.5.
2. The initial ranking value corresponding to a constraint with generality 1.0. This parameter is referred to as the *slope coefficient*. The tested values for the slope coefficient include 0.5 and 1.0.

Then the initial ranking value for each markedness constraint M is calculated using the following equation:

$$\begin{aligned}
\theta(M_{\text{init}}) &= 100\,(b + m g_M)\,, \\
\text{where } b &= y\text{-intercept coefficient} \in \{0.5, 1.0, 1.5\} \\
&\quad\ (\text{determines } \theta(\text{M}_{\text{init}}) \text{ for a constraint with } g_{\text{M}} = 0.0) \\
m &= \text{slope coefficient} \in \{0.5, 1.0\} \\
&\quad\ (\text{determines } \theta(\text{M}_{\text{init}}) \text{ for a constraint with } g_{\text{M}} = 1.0) \\
g_M &= \text{generality for constraint } M
\end{aligned} \tag{5.2}$$

For example, in the 5 000 randomly-sampled inputs of one North Seto learner's (Learner G, defined below in Section 5.4.3) observation stage, initial ranking values for two selected constraints are calculated as in (133) and (134):

(133) *B$_5$ was violated 6 346 times; therefore:

$$\begin{aligned}
g_{(*\text{B}_5)} &= 6\,346 \div 5\,000 = 1.2692 \\
\theta(*\text{B}_5) &= 100\,(b + m g_M) \\
&= 100\,(1.0 + 1.0\,(1.2692)) \\
&= 226.92
\end{aligned}$$

(134) $^*\underline{\mathrm{B}}_5\mathrm{F}_5$ was violated 861 times; therefore:

$$\begin{aligned} g_{(^*\underline{\mathrm{B}}_5\mathrm{F}_5)} &= 861 \div 5\,000 = 0.1722 \\ \theta(^*\underline{\mathrm{B}}_5\mathrm{F}_5) &= 100\,(b + mg_M) \\ &= 100\,(1.0 + 1.0\,(0.1722)) \\ &= 117.22 \end{aligned}$$

5.4.2.5 Random distribution function

Distribution Function 5 ($\mathscr{F}_5$): In addition to having a uniform distribution function $\mathscr{F}_1$ as a reference point against which to consider the success of the other, generality-based markedness distribution functions, I also ran simulations with markedness constraints randomly distributed over similar intervals as for $\mathscr{F}_4$.

The initial ranking value for each markedness constraint M is calculated using the following equation:

$$\begin{aligned} \theta(M_{\text{init}}) &= 100\,(b + mr_M)\,, \\ \text{where } b &= y\text{-intercept coefficient} \in \{0.5, 1.0, 1.5\} \\ m &= \text{slope coefficient} \in \{0.5, 1.0\} \\ r_M &= \text{simulated generality for constraint } M\text{, randomly sampled from } [0, 1] \end{aligned} \tag{5.3}$$

5.4.3 Simulation results - $\mathbf{M_{gen}} \gg \mathbf{M_{spec}}$

To demonstrate the effect of a general-over-specific markedness bias, I simulate acquisition of the three sample languages using Learner G, defined with the settings in Table 5.35. The choice of $\mathscr{F}_4$ for the distribution function is for illustrative purposes and was selected for its minimal requirement of *a priori* knowledge or calculation on the part of the learner, given that the initial distribution of markedness constraints can be determined based strictly on observation rather than prior analysis of set-theoretic relationships between classes of markedness constraints. Full results using learners with $\mathscr{F}_1$, $\mathscr{F}_2$, $\mathscr{F}_3$, and $\mathscr{F}_5$ (along with other values of b and m, in the cases of $\mathscr{F}_4$ and $\mathscr{F}_5$) are summarized in Appendix C, with discussion of the best performers in Section 5.5.

Learner G:	**Parameter**	**Setting**
	All basic parameters	Default
	Initial markedness values	$\mathscr{F}_4$ (input-calibrated): $b = 1.0, m = 1.0$

Table 5.35: Parameter settings for Learner G.

With only the general-over-specific markedness bias applied, the results do not initially appear to be any better than those of Learner A (default settings). Results are summarized in Table 5.36; results and final rankings for each individual language are discussed in more detail below.

A closer inspection of the ranking values for each language reveals that some changes are in fact present, even though their effects are masked in these summary results.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.37. Due to Learner G's lack of any of the other biases discussed thus far, the grammar acquired by the North Estonian learner is simply ruled

Language	Average rate of correct outputs (%)
North Estonian	24.55%
Finnish	26.18%
North Seto	29.82%

Table 5.36: Summary of results from simulations with Learner G.

by a high-ranking general faithfulness constraint. But contrary to several of the previous learners, the final grammar acquired by Learner G has the more general context-free markedness constraints $*F_3$ and $*B_2$ ranked significantly higher than the more specific no-disagreement constraints $*F_5\underline{B}_2$, $*F_5\infty\underline{B}_2$, $*\underline{B}_5F_3$, and $*\underline{B}_5\infty F_3$, which shows promise in terms of the potential for positional restrictions being enforced in their own right, rather than being misattributed to harmony.

Constraint	Final ranking value
Id(Bk)	254.00
$*F_3$	216.84
$*B_2$	204.84
$*B_1$	160.00
Id-σ_1(Bk)	154.00
$*F_5\underline{B}_2$	100.00
$*F_5\infty\underline{B}_2$	100.00
$*\underline{B}_5F_3$	100.00
$*\underline{B}_5\infty F_3$	100.00

Table 5.37: Excerpt of final ranking values for North Estonian after simulation with Learner G.

Finnish: Table 5.38 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. Similar to the North Estonian results, Id(Bk) is at the very top of the rankings. The $M_{gen} >> M_{spec}$ on its own does not seem to have any great effect on Learner G's attempted acquisition of the Finnish target grammar, which is of no particular concern because Finnish Learner F was already performing quite well and the main goal for this bias is to address the challenges involved in acquisition of North Estonian. At the very least, this bias does not appear to have made things any worse for the acquisition of Finnish.

North Seto: Table 5.39 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. Id(Bk) is at the top of this ranking as well. Similar to Finnish, this bias seems not to have made much difference to the North Seto learner, which again is of no great concern given North Seto Learner F's good performance and the aim of this bias being primarily to improve the North Estonian learner.

5.4.4 Simulation results - $F_{spec} >> F_{gen}$, promotion rate, $M_{gen} >> M_{spec}$

To demonstrate the combined effects of the *a priori* bias, Favour Specificity bias, tempered promotion rate, and general-over-specific markedness bias, I simulate acquisition of the three sample languages using Learner H, defined with the settings in Table 5.40.

Constraint	**Final ranking value**
Id(Bk)	254.00
$*B_2$	228.00
Id-σ_1(Bk)	178.00
$*F_3\underline{B}_5$	100.00
$*F_3\infty\underline{B}_5$	100.00
$*\underline{B}_5F_3$	100.00
$*\underline{B}_5\infty F_3$	100.00

Table 5.38: Excerpt of final ranking values for Finnish after simulation with Learner G.

Constraint	**Final ranking value**
Id(Bk)	254.00
Id-σ_1(Bk)	210.00
$*B_1$	167.50
$*F_4\underline{B}_5$	100.00
$*F_4\infty\underline{B}_5$	100.00
$*\underline{B}_5F_4$	100.00
$*\underline{B}_5\infty F_4$	100.00

Table 5.39: Excerpt of final ranking values for North Seto after simulation with Learner G.

Implementing all four of these modifications produces exceptional results, with even the North Estonian grammar now generating correct outputs in nearly every test evaluation. Results are summarized in Table 5.41; results and final rankings for each individual language are discussed in more detail below.

North Estonian: The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5.42.This grammar meets all the requirements for a target North Estonian grammar, though some pairs of ranking values are just close enough together that evaluation noise results in the odd swapped ranking; hence the 99.95% rate of correct outputs during testing.

Note that at first glance, these rankings may appear to be incorrect. In particular, $*F_5$ outranks Id(Bk), which would seem to suggest that all front vowels are banned from non-initial syllables. However, even though the rankings do not precisely match those in 5.1, the relationships between the front and back scale-referring constraints ensure that the correct patterns surface. This is due to the fact that $*B_2$ (which contains the back counterparts of the least-marked two front vowels) outranks $*F_5$. Tableaux (135) and (136) show that, no matter whether a non-initial vowel is in F_3, it will surface correctly due to the interaction of $*B_2$ and $*F_5$.

Learner H:	Parameter	Setting
	All basic parameters	Default
	A priori bias (F$_{\text{spec}}$ >> F$_{\text{gen}}$)	$d = 20$
	Favour Specificity bias (F$_{\text{spec}}$ >> F$_{\text{gen}}$)	Active
	Promotion rate	$1/w$
	Initial markedness values	$\mathscr{F}_4$ (input-calibrated): $b = 1.0, m = 1.0$

Table 5.40: Parameter settings for Learner H.

Language	Average rate of correct outputs (%)
North Estonian	99.95%
Finnish	100.00%
North Seto	100.00%

Table 5.41: Summary of results from simulations with Learner H.

(135) Sample evaluation of input /ø...e/ in the North Estonian grammar acquired by Learner H. The grammar successfully selects the faithful candidate, avoiding second-syllable vowels in set B$_2$. Candidates unfaithful in the first syllable have been omitted for the sake of simplicity, due to the high ranking of Id-σ_1(Bk).

/ø...e/	*B$_2$	*F$_5$	*F$_3$	Id(Bk)
☞ a. ø...e		**	*	
b. ø...ɤ	*!	*	*	*

(136) Sample evaluation of input /ø...y/ in the North Estonian grammar acquired by Learner H. The grammar successfully selects the candidate that avoids second-syllable vowels in set F$_3$. Candidates unfaithful in the first syllable have been omitted for the sake of simplicity, due to the high ranking of Id-σ_1(Bk).

/ø...y/	*B$_2$	*F$_5$	*F$_3$	Id(Bk)
a. ø...y		**!	**	
☞ b. ø...u		*	*	*

Finnish: Table 5.43 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. This grammar meets all the requirements for a target Finnish grammar (recall Figure 5.2), and the ranking values are far enough apart in value to behave effectively categorically, as evidenced by the 100% rate of correct outputs during testing.

North Seto: Table 5.44 shows the final ranking values for a selection of crucial constraints, after learning from simulated North Seto data. This grammar meets all the requirements for a target North Seto grammar (recall Figure 5.3), and the ranking values are far enough apart in value to behave effectively categorically, as evidenced by the 100% rate of correct outputs during testing.

5.4.5 Discussion

Applying a general-over-specific markedness bias to a learner's initial ranking values gives more general markedness constraints the opportunity to remain active in a grammar being acquired from

Constraint	Final ranking value
*B$_1$	116.40
*F$_5\underline{\text{B}}_2$	100.00
*F$_5\infty\underline{\text{B}}_2$	100.00
*$\underline{\text{B}}_5$F$_3$	100.00
*$\underline{\text{B}}_5\infty$F$_3$	100.00
Id-σ_1(Bk)	91.87
*F$_1$	82.84
*B$_2$	82.80
*F$_5$	74.59
*F$_3$	67.94
Id(Bk)	41.54

Table 5.42: Excerpt of final ranking values for North Estonian after simulation with Learner H.

Constraint	Final ranking value
*B$_2$	133.80
*F$_3\underline{\text{B}}_5$	100.00
*F$_3\infty\underline{\text{B}}_5$	100.00
*$\underline{\text{B}}_5$F$_3$	100.00
*$\underline{\text{B}}_5\infty$F$_3$	100.00
Id-σ_1(Bk)	91.09
Id(Bk)	57.40

Table 5.43: Excerpt of final ranking values for Finnish after simulation with Learner H.

positive data. For example, Table 5.45 (p. 112) shows both the initial and the final ranking values from North Estonian Learner H, for a selection of constraints. The more general constraints with higher initial ranking values have the opportunity to decrease slightly or fall quite drastically, depending on how often they contribute to learning errors (note the decrease of *F$_5\infty\underline{\text{B}}_5$ from 170.70 to −88.08, for example).

Without such a bias – if all markedness constraints start with the same value – it is possible for less-often violated, more specific markedness constraints to get credit for generating the attested patterns when more-often violated, more general constraints should in fact be the ones held responsible. North Estonian Learners F vs H provide a clear example of this pair of differing outcomes.

Of course, there are some situations in which the general constraints are truly too general to capture the patterns of a language. In these cases the errors caused by satisfying those general constraints trigger updates that demote them, allowing the more specific options to take precedence. The fact that a general-over-specific relationship is reversible given particular learning data is key to the success of this bias. For example, North Seto Learner H begins with *B$_5$ at 225.08 and *B$_1$ at 109.38. If these two constraints were to remain in the same relationship throughout the simulation, it would be impossible to acquire a grammar that bans only the vowels in B$_1$. However, by the end

Constraint	Final ranking value
$*F_4\underline{B}_5$	100.00
$*F_4\infty\underline{B}_5$	100.00
$*\underline{B}_5F_4$	100.00
$*\underline{B}_5\infty F_4$	100.00
Id-σ_1(Bk)	99.39
$*B_1$	89.38
Id(Bk)	56.60

Table 5.44: Excerpt of final ranking values for North Seto after simulation with Learner H.

of the simulation, although $*B_1$ has dropped to 89.38, $*B_5$ has fallen even further to 3.83, ensuring that $*B_1$ can do its job without interference from the more general constraint.

The combination of all of the biases introduced so far in this chapter – $F_{spec} >> F_{gen}$ (*a priori* and Favour Specificity), promotion rate, and $M_{gen} >> M_{spec}$, some of which have only been explored with a narrow set of parameters – produces learners that are able to acquire exceptionally successful grammars for all three of the sample languages. In Section 5.5 I present results from learners with a broader range of parameters than the ones specified in Sections 5.2, 5.3, and 5.4, and summarize the collections of settings that facilitate the most successful results.

5.5 Optimal learning conditions

Sections 5.2, 5.3, and 5.4 explored potential settings for the additional parameters and biases involved in addressing the various challenges encountered by an algorithmic learner with the basic settings described in Section 5.1. Learning simulations were run with a full crossing of all combinations of settings introduced throughout the chapter (1 080 combinations in total). Table 5.46 summarizes these parameters for the reader's convenience, and a full listing of the results is available in Appendix C.

The rates of correct outputs for all learning simulations are summarized by the histograms shown in Figure 5.7 (p. 114). The per-language ranges are 24.53% to 100.00% for North Estonian, 26.16% to 100.00% for Finnish, and 29.80% to 100.00% for North Seto. Although the minima and maxima are all quite similar from language to language, the distributions *within* those ranges are quite different, especially between North Estonian as compared to both Finnish and North Seto. North Estonian, with positional restrictions and no vowel harmony, has a larger portion of its results in the 60–90% range than the others do, and a much smaller portion in the 90–100% range. The histograms provide a very clear illustration of the fact that North Estonian is much more difficult for the learner to get exactly right.

In this section I examine the best-performing learners and analyze which combinations of parameters (and ranges of settings for each parameter) produce the best overall results. I also take care to consider the lowest-performing learners with any of these successful combinations of parameter settings, to ensure that the settings I deem crucial to success do – as consistently as possible – guarantee good results.

The success of a learner can be considered from more than one perspective. What follows is discussion of two different interpretations of what it means to be one of the best-performing learners.

Initial ranking values		**Final ranking values**	
*B_5	254.28	*B_1	116.40
*F_5	228.86	*$\text{F}_5\underline{\text{B}}_2$	100.00
*F_4	182.42	*$\text{F}_5\infty\underline{\text{B}}_2$	100.00
*$\text{F}_5\infty\underline{\text{B}}_5$	170.70	*$\underline{\text{B}}_5\text{F}_3$	100.00
*B_3	159.52	*$\underline{\text{B}}_5\infty\text{F}_3$	100.00
*$\text{F}_5\underline{\text{B}}_5$	153.64	Id-σ_1(Bk)	91.87
*$\underline{\text{B}}_5\infty\text{F}_5$	146.66	*F_1	82.84
*$\underline{\text{B}}_5\text{F}_5$	137.78	*B_2	82.80
*F_3	133.90	*F_5	74.59
*F_1	111.80	*F_3	67.94
*B_2	111.32	*F_4	52.15
*B_1	100.00	*B_5	43.14
*$\text{F}_5\underline{\text{B}}_2$	100.00	Id(Bk)	41.54
*$\text{F}_5\infty\underline{\text{B}}_2$	100.00	*B_3	30.55
*$\underline{\text{B}}_5\text{F}_3$	100.00	*$\underline{\text{B}}_5\infty\text{F}_5$	−52.16
*$\underline{\text{B}}_5\infty\text{F}_3$	100.00	*$\underline{\text{B}}_5\text{F}_5$	−55.33
Id-σ_1(Bk)	20.00	*$\text{F}_5\infty\underline{\text{B}}_5$	−88.08
Id(Bk)	0.00	*$\text{F}_5\underline{\text{B}}_5$	−105.14

Table 5.45: A selection of North Estonian Learner H's constraint ranking values at the beginning and the end of the learning simulation.

Section 5.5.1 summarizes which specific combinations of learning settings produce the absolute best results, whereas Section 5.5.2 addresses how generally (that is, with as few specifications as possible) a learner can be defined and still produce excellent results.

5.5.1 Best-performing individual learners

While none of the 1 080 simulations resulted in grammars that produce correct outputs 100% of the time for all three languages, there are a small number that achieve extremely close to it. Five of the learners tested acquired grammars that tested at 99.99% or better, for all three sample languages. The specifications for these learners are given in Table 5.47; their test results for each language are shown in Table 5.48. The learners in this set are labeled P, Q, R, S, T; however, these names are arbitrary and do not imply any particular relationship either to each other or to Learners A through H as described in Sections 5.1 through 5.4.

A few trends stand out when inspecting the characteristics of these five most successful learners:

(137) Initial markedness distribution:

a. Four of the five use $\mathscr{F}_4$ (the input-calibrated general-markedness distribution), with the fifth using $\mathscr{F}_5$ (which also distributes markedness constraints through a range of initial ranking values, albeit randomly).

b. Four of the five use the same y-intercept and slope settings for creating the distribution ($b = 1.5, m = 0.5$).

(138) Four of the five leverage the Favour Specificity bias.[48]

[48] On the other hand, only one of these top five learners – Learner P – omits the Favour Specificity bias; however, it

Parameter	**Available settings**
All basic parameters	Default
A priori bias ($F_{\text{spec}} >> F_{\text{gen}}$)	None or $d \in \{0, 10, 20, 30, 40\}$
Favour Specificity bias ($F_{\text{spec}} >> F_{\text{gen}}$)	Active or inactive
Promotion rate	1 (default) $l/(1+w)$ $l/(l+w)$ $1/w$ $1/(1+w)$
Markedness distribution function	$\mathscr{F}_1$ (uniform) starting at 100, 300, or 500 $\mathscr{F}_2$ (stratified by constraint type) $\mathscr{F}_3$ (stratified by stringency set), either top-down or bottom-up $\mathscr{F}_4$ (input-calibrated), $b \in \{0.5, 1.0, 1.5\}, m \in \{0.5, 1.0\}$ $\mathscr{F}_5$ (random), $b \in \{0.5, 1.0, 1.5\}, m \in \{0.5, 1.0\}$

Table 5.46: All available parameter settings for learners described in this chapter.

	***a priori* bias (d)**	**Favour Specificity**	**Promotion rate**	**Markedness distribution**	
				Function	**Details**
Learner P	40	inactive	$1/(1+w)$	$\mathscr{F}_4$	$b = 1.0, m = 0.5$
Learner Q	0	active	$l/(l+w)$	$\mathscr{F}_4$	$b = 1.5, m = 0.5$
Learner R	30	active	$1/w$	$\mathscr{F}_4$	$b = 1.5, m = 0.5$
Learner S	none	active	$l/(l+w)$	$\mathscr{F}_4$	$b = 1.5, m = 0.5$
Learner T	40	active	$l/(l+w)$	$\mathscr{F}_5$	$b = 1.5, m = 0.5$

Table 5.47: Specifications for the five learners whose final grammars achieve a 99.99% success rate.

(139) Three of the five use a promotion rate of $l/(l+w)$. Promotion rate does vary somewhat but conspicuously does not include either $l/(1+w)$ or default (1).

To illustrate the overall contribution of these characteristics to successful learning, consider the *lowest* success rates (shown in Table 5.49) for any learners whose settings include $\mathscr{F}_4$ with $b = 1.5, m = 0.5$; Favour Specificity; and promotion rate of $l/(l+w)$ (Type 2), $1/w$ (Type 3), or $1/(1+w)$ (Type 4).

If the promotion rate is restricted to only $l/(l+w)$ (Type 2), as is shared by three of the five best learners, the lower bound for success becomes even better, as shown in Table 5.50.

These results show that even when allowing the *a priori* bias to vary, this combination of three

is not coincidence that the *a priori* bias is set quite high for that learner. Both of these biases help to create distance between the faithfulness constraints, so it stands to reason that if one is left out, the other will have to take on all of that work.

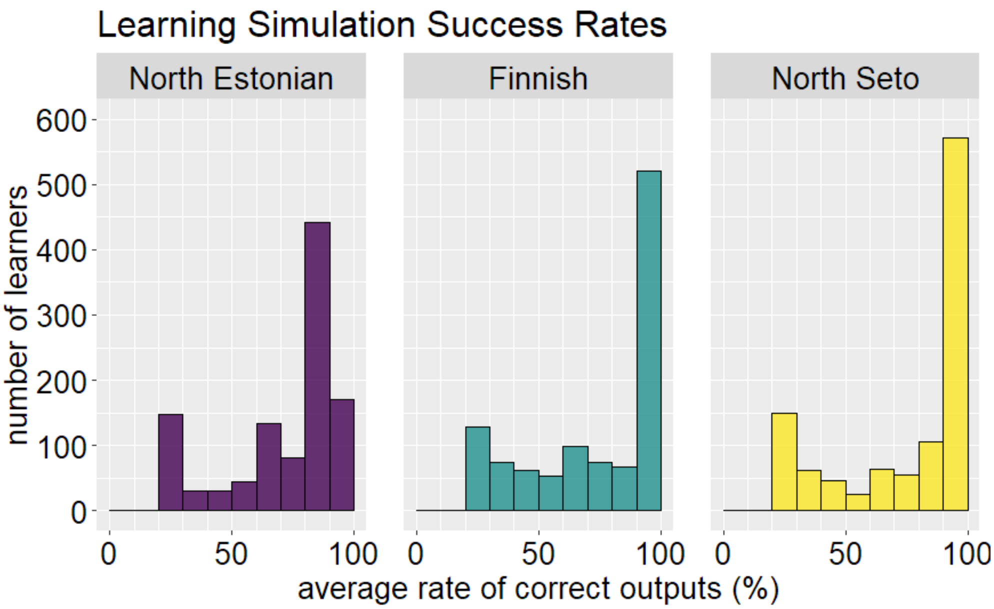


Figure 5.7: Distribution of success rates for all learning simulations (1 080 parameter combinations for each of 3 languages).

	North Estonian	Finnish	North Seto
Learner P	99.99%	100.00%	99.99%
Learner Q	99.99%	100.00%	100.00%
Learner R	99.99%	100.00%	99.99%
Learner S	99.99%	100.00%	99.99%
Learner T	99.99%	99.99%	100.00%

Table 5.48: Results for the five learners whose final grammars achieve a 99.99% success rate.

specifications dependably produces excellent results for Finnish and North Seto, and reasonably good results for North Estonian. If promotion rate is fixed as well, the North Estonian results join the others in near-perfect consistency. Therefore choosing settings that align with those in 137, 138, and 139 defines a learner that is likely to produce excellent results.

It must be acknowledged that the 99.99% benchmark is somewhat arbitrary. While each of the five learners at or above that cutoff uses $\mathscr{F}_4$ to determine initial markedness constraint values, it is also true that relaxing the cutoff broadens the range of settings seen in the top-performing learners. For example, lowering the benchmark slightly, to 99.9%, increases the number of top-performing learners to 71. General trends are summarized in Table 5.51, showing observed vs expected values for various parameter settings appearing in the top-performing learners. Calculations for expected values are shown in Appendix A.2.

From Table 5.51 we can conclude that, broadly, the parameter settings that are more likely to contribute to a learner's success are a greater *a priori* bias ($d \in \{30, 40\}$); an active Favour Specificity bias; a promotion rate $\in \{l/(l+w), 1/w\}$; and/or a markedness distribution function that is either uniform, stratified by constraint type, or calculated based on observed inputs.

Language	**Average rate of correct outputs (%)**
North Estonian	83.62%
Finnish	100.00%
North Seto	99.95%

Table 5.49: Summary of *worst* results from simulations with learners defined based on trends from Learners P, Q, R, S, T. Learners represented in this table have promotion rates of Types 2, 3, and 4.

Language	**Average rate of correct outputs (%)**
North Estonian	98.12%
Finnish	100.00%
North Seto	99.98%

Table 5.50: Summary of *worst* results from simulations with learners defined based on trends from Learners P, Q, R, S, T. All learners represented in this table have a promotion rate of Type 2.

Parameter	**Attested settings in top 71 learners**		
	Setting	**# observed**	**# expected**
A priori bias	None	8	11.83
	$d = 0$	8	11.83
	$d = 10$	9	11.83
	$d = 20$	10	11.83
	$d = 30$	17	11.83
	$d = 40$	19	11.83
Favour Specificity bias	Active	68	35.50
	Inactive	3	35.50
Promotion rate	1 (default)	0	14.20
	$l/(1+w)$	0	14.20
	$l/(l+w)$	29	14.20
	$1/w$	28	14.20
	$1/(1+w)$	14	14.20
Markedness distribution	$\mathscr{F}_1$	15	11.83
	$\mathscr{F}_2$	8	3.94
	$\mathscr{F}_3$	8	7.89
	$\mathscr{F}_4$	28	23.67
	$\mathscr{F}_5$	12	23.67

Table 5.51: Observed vs expected values for parameter settings appearing in the learners with success rates greater than 99.9%. Highlighted rows indicate observed > expected.[49]

[49] Although observed > expected for $\mathscr{F}_3$, the row is not highlighted as the difference is not large enough to be particularly meaningful.

There are some more subtle relationships among these settings that can be further teased apart. First, with respect to the selection of a promotion rate, the three options appearing in the top five learners specified in Table 5.47 are $l/(l+w)$, $1/w$, and $1/(1+w)$. In fact, these are the *smallest* three promotion functions (proofs in Appendix A.1.2). Not only that, but the interaction of all promotion rate types with *a priori* d values can be observed from the heat map in Table 5.52. The distribution of the top 71 learners within this space suggests that less stringent *a priori* biases produce better results when paired with $l/(l+w)$, Type 2, but that the larger *a priori* biases perform better when paired with $1/w$ or $1/(1+w)$, Types 3 or 4.

		a priori (d)					
		no	0	10	20	30	40
Prom. rate	1 (default)	0	0	0	0	0	0
	$l/(1+w)$	0	0	0	0	0	0
	$l/(l+w)$	7	6	6	4	4	2
	$1/w$	1	2	3	4	8	10
	$1/(1+w)$	0	0	0	2	5	7

Table 5.52: Combinations of promotion types and *a priori* bias values observed in the top 71 learners (99.9% or better on all three sample languages).

Second, within the baseline set of uniform initial ranking values $\mathscr{F}_1$, starting values of 300 and 500 appear more often than expected, and starting values of 100 appear less often than expected. This suggests that even if all markedness constraints start with the same value, the simulation is more likely to succeed if that uniform value is further from the faithfulness constraints.

Finally, consider that there are two main categories of markedness distribution functions: those that take generality into account ($\mathscr{F}_2$, $\mathscr{F}_3$, and $\mathscr{F}_4$) and those that do not ($\mathscr{F}_1$ and $\mathscr{F}_5$). Grouped together in this way, the ones that consider generality are observed in $8 + 8 + 28 = 44$ cases (and expected in 35.5), whereas the ones that do not are observed in $15 + 12 = 27$ cases (and expected in 35.5). Thus zooming out of the individual function types illuminates the fact that incorporating generality into the initial markedness constraint distribution is also a useful tool.

These results show that, in broad terms, a larger *a priori* bias is a helpful parameter to include in a learner acquiring Finnic vowel patterns, as is the Favour Specificity bias, and a promotion rate of $l/(l+w)$ (Type 2) or $1/w$ (Type 3). Several different initial markedness constraint distribution options contribute to learner success; those that initially distribute markedness constraints according to generality tend to result in acquisition of better final grammars.

Section 5.5.2 considers in a more structured way which generalized combinations of settings are overall more likely to produce successful results.

5.5.2 Best-performing sets of learners

A slightly different perspective from the one offered in Section 5.5.1 is to ask not only which settings are common to the most-successful learners, but which combinations of settings can be generalized as the most crucial to successful learning. For this, rather than keeping the benchmark for success as high as in Section 5.5.1, I relax it slightly further, to 99%, in order to have more data from which to attempt to draw generalizations. From the 95 learners that meet this benchmark, I look for commonalities in their settings and present the sets of successful learners with the fewest specified parameters;[50] see Table 5.53. The range of results for each language under each of these minimal generalizations are summarized in Table 5.54.

	Favour Specificity	**Promotion rate**	**Markedness distribution**		***a priori* bias (d)**
			Function	**Details**	
Set X	active	$l/(l+w)$	$\mathscr{F}_1$	300 or 500	any
Set Y	any	$1/(1+w)$	$\mathscr{F}_4$	$b = 1.5, m =$ any	40

Table 5.53: Most broadly-encompassing sets of settings drawn from learners with results over 99%.

	North Estonian		**Finnish**		**North Seto**	
	Best	**Worst**	**Best**	**Worst**	**Best**	**Worst**
Set X	100.00%	99.93%	100.00%	100.00%	100.00%	99.93%
Set Y	99.99%	99.45%	100.00%	99.99%	100.00%	99.96%

Table 5.54: Best- and worst-case results for both sets of learners from Table 5.53

The information summarized in Tables 5.53 and 5.54 demonstrates that there are two sets of learners with comparatively minimal specifications, that produce results consistently better than 99%.

Considering *sets* of learners in this way, rather than the *individual* learners as in Section 5.5.1, provides some insight into the ways that the parameters interact with each other and with the learning process more generally.

Set X demonstrates that *a priori* bias is moot when it comes to a learner's ability to acquire a target grammar, assuming that Favour Specificity is active, the promotion rate is $l/(l+w)$ (Type 2), and markedness constraints all start quite high (distributed by $\mathscr{F}_1$ at 300 or 500). This combination of parameters suggests that we can avoid considering generality of markedness constraints, as long as (a) markedness constraints start far enough away from faithfulness constraints, (b) specific faithfulness constraints are given the opportunity to rise on their own, independent of their general counterparts, and (c) the promotion rate is not too close to zero.

Set Y, on the other hand, shows that the application (or lack thereof) of Favour Specificity is moot, as long as the promotion rate is $1/(1+w)$ (Type 4), the *a priori* bias has $d = 40$, and initial values of the markedness constraints are assigned based on generality (distributed by $\mathscr{F}_4$ with $b = 1.0$). This combination of parameters suggests that Favour Specificity need not be specified, as long as the *a priori* bias is set high enough and the promotion rate is very small.

Both sets of learners assign initial ranking values to markedness constraints that are much higher above the initial faithfulness constraint values than would be the case in a learner with default settings. Combined with the cautious promotion rates, this provides the learner with plenty of time and space to allow the markedness constraints to establish their relative rankings before the faithfulness constraints rise to the top.

[50] In other words, I ask how large a class of learners I can define – using the fewest specifications – and still have all of the learners in that class achieve a success rate of at least 99%.

5.6 General discussion and conclusion

In this chapter, I introduced three main categories of learning biases that contribute to the success (or lack of it) of a GLA-type learner acquiring a range of Finnic vowel patterns from North Estonian, Finnish, and North Seto. The first category was that of specific-over-general faithfulness bias, including *a priori* bias and Favour Specificity, the latter of which was novelly adapted for use in online learning. The second category addressed the update rule's promotion rate. The third was a novel implementation of a general-over-specific markedness bias, which included options such as stratified (determined from constraint structure) and input-calibrated (determined from observed learning data) initial distributions of markedness constraints. These biases are discussed in more detail below.

***A priori* bias**: The higher *a priori* bias values of $d = 30$ or 40 that turned out to be prevalent in the most-successful learners mean that each specific faithfulness constraint has a value high enough above its general counterpart such that their relationship during evaluation is effectively categorical. This ensures that any vowels in privileged positions (in this case, the initial syllable) are in fact more likely to remain faithful to their underlying values than vowels elsewhere in the word. For the Finnic languages, this is absolutely necessary as full contrast (with respect to vowels) in any given language is only guaranteed in the first syllable. On its own, the *a priori* bias inevitably results in each pair of faithfulness constraints moving in lockstep at a distance of no less than 30 (or 40) apart, since there will always be at least as many faithfulness errors made in the word as a whole as in the first syllable.

Favour Specificity: The Favour Specificity bias was originally proposed by Hayes (2004) for a batch learner (Low-Faithfulness Constraint Demotion), and I have presented a novel adaptation of it to an online learning context. An active Favour Specificity bias works in concert with the *a priori* bias to prioritize specific faithfulness constraints over their general counterparts. This particular setting is implemented via the learner promoting only the specific version of a faithfulness constraint when adjusting after an error made in which both the specific and general versions are eligible to be promoted. Such a bias comes with the potential for each specific faithfulness constraint to rise further above its general counterpart than what the *a priori* bias requires, since it will not always be the case that all faithfulness errors result in promotion of the general constraint. In the Finnic languages, this kind of additional space is important for languages with positional restrictions, which require context-free markedness constraints against particular sets of vowels to be ranked between specific and general faithfulness constraints.

Promotion rate: The $l/(l+w)$ (Type 2), $1/w$ (Type 3), and $1/(1+w)$ (Type 4) promotion rates that appeared in the most-successful learners all fall in the interval $(0, l/w)$ that Magri (2012) showed to result in efficient convergence of a GLA-type learner. These promotion rates also share a second important characteristic, which is that they are guaranteed to produce fractions ≤ 1. This is crucial for ensuring that promotion amounts are no greater than the base plasticity, in order to remain conservative when giving credit to winner-preferring constraints (the Credit Problem; Dresher, 1999).

$\mathbf{M_{gen}} \gg \mathbf{M_{spec}}$: The general-over-specific markedness bias was originally proposed by Albright and Hayes (2006) for a Minimal Generalization Learner that tracks constraint generality during a constraint induction phase. I have presented a novel implementation is input-calibrated and can be used by a learner that is presented with already-formulated constraints. Markedness constraints with initial values distributed by generality help to ensure that more general constraints are given the opportunity to take credit for a grammar's phonotactics before specific ones do. These initial distributions can be determined either by prior analysis of a constraint's structure (whether its type or the stringency sets that it references) or by calculating its application rate from a set of learning inputs. The latter approach, dependent only on tracking constraint violations in observed data, is arguably more ecologically valid than the version that requires prior theoretical analysis.

Although all of these approaches have been previously proposed in some form or another, my use of them in this context both introduces some novel implementations and requires (permits) the biases to work in concert to define learners that acquire final grammars that nearly perfectly replicate the targets. The novel implementations that I presented included: (a) a version of Hayes's (2004) *Favour Specificity* principle that can be applied to an online rather than a batch learner; (b) promotion rates Types 2 and 3, which provide alternative methods for being conservative with respect to the Credit Problem; and (c) markedness distribution functions $\mathscr{F}_2$, $\mathscr{F}_3$, and $\mathscr{F}_4$ as options for prioritizing general markedness constraints over specific ones. All three types of these new proposals provide a crucial service to the learner, which is to ensure that it acquires a maximally restrictive grammar (Hayes, 2004; Tessier, 2007) and thus avoids falling into the Subset Problem (Angluin, 1980; Baker, 1979).

Section 5.5 summarized the most successful out of a total of 1 080 learners with different parameter combinations. The wide range of combinations means that there are several potential interactions among the parameter settings that could lead to dependably good results. This kind of information would be extremely useful in planning future learning simulations, particularly in terms of being able to reduce the hypothesis space and prioritize computing time for learners more likely to succeed. However, beyond the general trends discussed in Section 5.5, detailed analysis of the impact of each variable (or each combination) is outside the scope of this dissertation. Such tasks are left for future research.

I conclude this section by revisiting an issue mentioned briefly in Section 3.2.1: the potential advantages of including MaxIO in the constraint set and therefore deletion as an additional repair strategy for avoiding disharmonic vowel sequences. In Section 5.2.4 I describe the oscillation of antagonistic (or near-antagonistic) context-free markedness constraints, which prevents them from dropping out of the way so that the constraints that should be active in the target grammar can indeed be active. The reason for the oscillation is that the only possible repair for markedness violations is to change the backness of a vowel; therefore avoiding a violation of (e.g.) $*F_5$ means incurring a violation of both Id(Bk) and $*B_5$. When this obstacle first became clear, I ran some pilot simulations including MaxIO and MaxIO-σ_1 in the constraint set, and deletion options in the candidate sets. My hope in doing so was that the oscillating constraints might be decoupled by providing repair options that did not only ever change vowel backness.

Unfortunately, while this approach may seem to work from a narrow trial-by-trial perspective, it does not appear to help in a broader sense (i.e., through an entire learning simulation). At the beginning of a learning simulation, deletion candidates are selected as optimal the overwhelming majority of the time, because deleting a vowel (violating an initially low-ranked faithfulness constraint) allows the current grammar to avoid *many* different initially high-ranked markedness violations, of both context-free and no-disagreement constraints. In errors of this kind, $*F_5$ and $*B_5$ (for example, among many others) are not forced to have opposite winner vs loser preferences. The small subset of a Finnish ERC matrix in 140 illustrates.

(140) ERC matrix demonstrating that *F$_5$ and *B$_5$ can move independently of each other when the loser is a deletion candidate.

input	*winner* ∼ *loser*	*F$_5$	*B$_5$	Id(Bk)	MaxIO
/æ...i...æ/	æ...i...æ ∼ æ...i..._	L	e	e	W
/æ...i...ɑ/	æ...i...ɑ ∼ æ...i..._	e	L	e	W

On the other hand, since almost all of the deletion-based errors happen at the beginning of the simulation, the separation of antagonistic constraints is short-lived: the learner soon figures out that deletion is not an appropriate tactic. It avoids violating the Max constraints, and in doing so begins to violate the Ident constraints instead, leaving the antagonistic constraints to oscillate as usual for remainder of the simulation (see 141).

(141) ERC matrix demonstrating that *F$_5$ and *B$_5$ return to preferring opposite candidates once MaxIO is no longer being violated.

input	*winner* ∼ *loser*	*F$_5$	*B$_5$	Id(Bk)	MaxIO
/æ...i...æ/	æ...i...æ ∼ æ...i...ɑ	L	W	W	e
/æ...i...ɑ/	æ...i...ɑ ∼ æ...i...æ	W	L	W	e

It is certainly *possible* for the problematic constraints to drop far enough, and fast enough, to facilitate the desired interactions between the ideal top-ranked markedness constraints and the Ident constraints (for example, in pilot Finnish learning simulations). However, such movement is not guaranteed, as was the case in pilot North Estonian simulations where many markedness constraints remained crowded near their initial values, paving the way for faithfulness constraints to rise too far (see Table 5.55). This persistent challenge means that introducing deletion as an alternate repair is likely not a good strategy for use as a more widely-applicable tool.

Constraint	**Final ranking value**
MaxIO-σ_1	114.00
MaxIO	112.00
Id-σ_1(Bk)	112.00
*B$_1$	100.00
*$\underline{B}_5F_3$	100.00
*$\underline{B}_5\infty F_3$	100.00
*$F_5\underline{B}_2$	100.00
*$F_5\infty\underline{B}_2$	100.00
Id(Bk)	92.00
*F$_1$	92.00
*B$_2$	92.00
*F$_3$	76.00

Table 5.55: Excerpt of final ranking values for North Estonian after simulation with pilot deletion learner.