This is Chapter 4/5 only. Refer to 20240818 dissertation outline and notes for complete outline.

Kaili's very official dissertation draft

For purposes of graduating in the near-ish future

4/5. Learning biases and optimal learning conditions

Yellow highlighting indicates updates/follow-up needed.

Notes highlighted in blue throughout.

Prior to this chapter, I will have introduced:

- The balto-finnic languages / vowel pattern typology
- Representative sample languages (patterns) for the above
- My constraint set & OT analysis of the sample languages
- Learning algorithms, in particular GLA-type learners
    - Definition of idempotence
    - Definition of θ-notation for ranking values
    - Explain my method of calculating "average frequency of correct results"
- My methods for simulations - python script etc
- Simulated learning data for the sample languages

KCH blue highlighting = just for my reference
green = something maybe you should highlight?
yellow = commented on

as someone less familiar with this domain, I think it would help if you were slightly more specific here — that is, I think that what you mean is something like “if you’re going to simulate acquisition, you have to decide on the settings for many variables, parameters, and biases” (rather than “there are variables, parameters, and biases that exist in the real world that you have to know about” or “there are variables, parameters, and biases that these systems come pre-programmed with that you have to understand the effects of” etc.)

Simulating acquisition of a grammar with an algorithmic learner involves many potential variables, parameters, and biases. In this chapter I introduce those factors that are relevant to learning Balto-Finnic vowel patterns, and discuss the impact that each has on the sample languages Finnish, North Estonian, and North Seto.

in what sense? e.g., you’re not personally going to change them within the context of the thesis? or you actually think that in {these languages, all languages} they do not change? and is that change over time? or something else? I’m sure you’ll explain in the section, but giving a brief hint at the scope here would be helpful.

In Section 4.1 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. Sections 4.2, 4.3, and 4.4 introduce various biases, each of which contributes to solving one of the problems previously identified while simultaneously uncovering more subtle obstacles not previously apparent. Section 4.2 investigates the specific-over-general faithfulness bias; Section 4.3 focuses on options for varying the promotion rate applied at each learning update; and Section 4.4 explores the general-over-specific markedness bias. Finally, in Section 4.5, 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 Balto-Finnic languages.

4.1. Learning simulations with default settings

The GLA, as specified by Boersma and Hayes (2001) and discussed in Section 3.X, 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.

…“as described there”? (so, you won’t be rehashing them here? and does 3.X also give rationale for that approach?)

4.1.1. 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.

maybe bold or otherwise highlight each of the biases for easy identification and reference?

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.[1] 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), Smolensky (1996), Hayes (2004), Prince and Tesar (2004), and Jesney and Tessier (2011). 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.

[1] 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.

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

emphasize for clarity?

lol…clearly we had the same idea…I think I'd do the emphasis on first mention though

third?

The fourth 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 evaluation) did not affect the overall ranking.

and did 20,000 come from anywhere? is it standard? did you discover it by trial and error?

fourth?

The fifth parameter that will remain constant is the permission of negative ranking values. 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 {θ_{C1} = 100, θ_{C2} = 50} produce the exact same ranking as the values {θ_{C1} = -25, θ_{C2} = -75}. Given this fact, the default OTSoft (Hayes et al, 2013a) approach for GLA learning is used; that is, to permit demotion of constraints even when the resulting ranking value is negative. [I was sure that I'd seen someone else make the argument that negative values are ok when using GLA with classic OT, but I can't for the life of me find it. Does this ring a bell for anyone else? I'm happy to explain it myself but it seems silly to do so if it already exists elsewhere!]

"as discussed in §X.X"

something seems a little funny in the overall framing of this section — are you truly 'assuming' these remain constant, or is it the case that you did some initial pilot testing of all of these parameters, and based on not seeing any effects, you are setting them to be constant from here on out? 'Assuming' sounds to me like you have a priori beliefs that they should be constant, but doing the pilot testing makes it seem like you were not totally sure and might have made a different choice if the pilots had gone differently.

The last few parameters that are held constant across simulations are the organization of learning trials into stages, evaluation noise, and the plasticity function. None of the results I discuss appear to depend on changes to these settings, so the default OTSoft (Hayes et al, 2013a) assumptions for GLA learning are used; they are summarized in Table 1.

as someone not super versed in this area, I'll need these defined at some point — perhaps that came earlier in the GLA intro?

Parameter	**Stage 1**	**Stage 2**	**Stage 3**	**Stage 4**
Number of learning trials	5000	5000	5000	5000
Evaluation noise	2	2	2	2
Plasticity	2	0.2	0.02	0.002

Table 1. Invariant settings for GLA learning.

4.1.2. Simulation results

Initial learning simulations use all of the basic parameters (those described in Section 4.1.1) at their default settings, with no additional biases or parameters introduced. Learner A is defined with the settings in Table 2.

Parameter	Setting
All basic parameters	Default

Table 2. Parameter settings for Learner A.

Under these default conditions, learning simulations for all three sample languages fail to acquire the target grammars, producing instead fully-faithful grammars. Such fully faithful grammars succeed 100% of the time on the input forms, since we assume idempotence. However, they are very poorly equipped to deal with illicit test forms. Test results are summarized in Table 3.

what does ‘illicit’ mean here? nonce words? or inputs that are specifically not following expected grammatical patterns?

Language	Average frequency of correct outputs
Finnish	0.2654
North Estonian	0.2455
North Seto	0.2982
Overall	0.2697

Table 3. Summary of results from simulations with Learner A.

specifically of illicit / nonce words?

(remind us of the section these were defined in for reference)

Table 4 shows the final ranking values for a selection of crucial constraints, after learning from simulated Finnish data. Id(Bk) has risen to the top of the rankings, and even though its distance from *B_2, the relevant no-disagreement constraints, and others with values near 100 is small enough that evaluation noise might cause it to swap rankings with one of its neighbours, the grammar does not align with the crucial rankings proposed in Section 3.X:

at some point will you discuss what the difference needs to be for this to happen?

just make it slightly clearer within the prose that it Id(Bk) was “supposed” to be crucially *below* all these other constraints

*B_2, *$F_3…\underline{B}_5$, *$F_3\underline{B}_5$, *$\underline{B}_5…F_3$, *$\underline{B}_5F_3$, Id(Bk)σ_1 >> Id(Bk)

…and more generally, will you be discussing what it means to be ‘unranked’ here — does this have to correspond with exactly the same ranking value (e.g. 106 and 106) or is there some range that ‘counts’ as being ‘unranked’?

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

kind of related to the above comments, I’m wondering if there’s a clearer way to present the rankings and the comparison to the ideal rankings…e.g. at the least, can both be presented vertically or both horizontally? and then e.g. if it’s vertically, there are horizontal lines between groups of constraints that are considered ‘unranked’ within the group? and then can all ‘unranked’ constraints within a group be presented in the same order across rankings (e.g. always alphabetically) for easy comparison?

Table 4. Excerpt of final ranking values for Finnish after simulation with Learner A.

The final ranking values for a selection of crucial constraints, after learning from simulated North Estonian inputs, are shown in Table 5. Id(Bk) has risen to the top of the rankings, and even though its distance from *B_1 (and other constraints with values near 100) is small enough that evaluation noise might cause it to swap rankings with one of its neighbours, the crucial rankings Id(Bk)σ_1 >> *F_3, *B_2 >> Id(Bk), proposed in Section 3.X, are certainly not achieved.

again, can you spell out the discrepancies a bit more? e.g. "…since Id(Bk)syll1, which needs to be at the top, and Id(Bk), which needs to be at the bottom, are in exactly the opposite order here"

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

(similar to above presentation comments)

Table 5. Excerpt of final ranking values for North Estonian after simulation with Learner A. A selection of no-disagreement constraints are included here for the purpose of comparison with results of simulations discussed in subsequent sections.

Learning from simulated North Seto data results in final ranking values for a selection of crucial constraints shown in Table 6. Id(Bk) has risen to the top of the rankings, and even though its distance from *B_1, the relevant no-disagreement constraints, and others with values near 100 is small enough that evaluation noise might cause it to swap rankings with one of its neighbours, the grammar does not meet the crucial target rankings proposed in Section 3.X:

*F_4…$\underline{B}_5$, *$F_4\underline{B}_5$, *$\underline{B}_5$…F_4, *$\underline{B}_5F_4$, Id(Bk)syl1 >> *B1 >> Id(Bk)

(similar comments as above re: making it clearer in the text and possibly presenting the comparison results more clearly)

Constraint	Final ranking value
Id(Bk)	116
*F_4…$\underline{B}_5$	106
*$\underline{B}_5$…F_4	106
*$F_4\underline{B}_5$	106
…	…
*B_1	102
…	…
*$\underline{B}_5F_4$	100
…	…
Id(Bk)σ_1	80

Table 6. Excerpt of final ranking values for North Seto after simulation with Learner A.

4.1.3. Discussion and challenges

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 4.1.2, 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.

both of these phrasings make it sound like you will discuss multiple issues in this section, but currently it's actually the only one — it's fine if it's the only one, just make it clearer from the start that this is what's happening

With respect to the results presented in Section 4.1.2, the most glaring problem is that Id(Bk) is highest ranked in all three. 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.

The reason for Id(Bk)'s rise all the way to the top of the rankings is due to a lack of *a priori* relative ranking of the specific and general faithfulness constraints (specifically, an obligation for Id(Bk)σ_1 to outrank Id(Bk)) combined with the 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 ERC matrix in Table 7 shows that for a surface form of /o..ɑ/ (grammatical in all three of the sample languages), Id(Bk) is a winner-preferrer for *any* error and Id(Bk)σ_1 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(Bk)σ_1 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.

sorry, what does ERC stand for?

input	candidates	markedness constraints	Id(Bk)	Id(Bk)σ_1
/o..ɑ/	o..ɑ ~ o..æ	…	W	
/o..ɑ/	o..ɑ ~ ø..ɑ	…	W	W
/o..ɑ/	o..ɑ ~ ø..æ	…	W	W

Table 7. ERC matrix demonstrating that under assumption of idempotence, all learning errors have Id(Bk) as a winner-preferring constraint.

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 4.1, it must be addressed before any others can be revealed. Section 4.2 presents a solution for this problem.