Learning Parametric Stress without Domain-Specific Mechanisms

Aleksei Nazarov and Gaja Jarosz
Harvard University and University of Massachusetts Amherst

1 Introduction

The Principles and Parameters (P&P) approach to language typology and acquisition (Chomsky 1981/1982) has provided important insights in both syntactic and phonological theory. In phonology, parametric models exist in the realms of word stress (Halle and Vergnaud 1987, Dresher and Kaye 1990, Hayes 1995), consonant assimilation (Archangeli and Pulleyblank 1994, Cho 1999), and syllable structure (see, e.g., Blevins 1995). In this paper, we address two fundamental questions about learnability within the P&P approach: how might a learner acquire the parameter settings for their target language? What capabilities must we ascribe to learners in the P&P framework – in particular, could domain-general statistical learning strategies be sufficient? The answers to these questions have broad-reaching implications about the nature and exact content of Universal Grammar (UG). They also complement theoretical work within and across frameworks by enriching understanding of various theoretical frameworks' computational properties. Our focus is on the P&P approach to metrical phonology; however, the learning model we introduce is broadly applicable to parametric theories in phonology and beyond. Therefore, our results and conclusions have implications for learning in P&P more generally. The main thrust of our argument is that learning in P&P metrical phonology can be more successful with weaker assumptions about UG than previous work has claimed. This simultaneously strengthens the computational motivations for the P&P approach and weakens the argument for domain-specific learning mechanisms.

It is useful to compare and contrast the computational properties of competing theoretical frameworks, such as P&P and Optimality Theory (OT; Prince and Smolensky 1993/2004). By contributing to the development of explicit learning models in various frameworks, we are constructing tools that may help uncover or highlight differences in predictions. Typological and learning predictions can often be difficult to foresee without explicit implementations. Conversely, computational analyses of theoretical frameworks can also uncover deep similarities between divergent approaches (for an overview of such results in phonology, see Heinz 2011a,b). Our focus is on the need for domain-specific mechanisms in phonological learning, and our findings for P&P align with much recent work on learning in constraint-based frameworks, providing converging support for this conclusion. From the perspective of model development, studying a learning problem from the perspective of one framework can often lead to insights about learning in other frameworks. Indeed, the learning model we propose for P&P builds on a recent proposal for learning in probabilistic OT (Jarosz 2015). Whatever one's theoretical persuasion, we therefore have to consider the implications of both OT and P&P theories of stress for learning, typology, and Universal Grammar.

We will be responding to recent work by Pearl (2007, 2008, 2009a,b, 2011) on the learning of stress parameters. In contrast to OT, where existing models rely only on domain-general learning strategies (see, e.g., Tesar 1995, Tesar and Smolensky 2000, Boersma 1997, Boersma and Hayes 2001, Boersma and Pater 2016, Jarosz 2013, 2015), Pearl argues that P&P approaches to UG must be supplemented with domain-specific learning mechanisms (as exhibited in earlier P&P learning work, e.g., Dresher and Kaye 1990, Gibson and Wexler 1994, Berwick and Niyogi 1996, Lightfoot 1999). We will argue instead that, even if one presupposes that UG encodes phonological grammars using the P&P framework, UG does not need to

* We would like to thank Adam Albright, Gašper Beguš, Naomi Feldman, Edward Flemming, Bruce Hayes, Michael Kenstowicz, Armin Mester, Erin Olson, Joe Pater, Ezer Rasin, Juliet Stanton, Kristine Yu, Sam Zukoff, and audiences at the UMass Amherst Sound Workshop, the MIT Phonology Circle, NECPhon 10 at UMass Amherst, AMP 2017 at USC, and the LSA 2017 Annual Meeting for their feedback that has greatly benefited this work. All errors remain our own.

Proceedings of *AMP 2016*

provide a theory of learning (i.e., so-called domain-specific learning mechanisms) in addition to a theory of grammar. The solution we offer is based on insights from Jarosz's (2015) learner for OT, and diverges minimally from Yang's (2002) Naïve Parameter Learner, while maintaining its advantages (online, gradual learning of a probabilistic parameter grammar, in addition to learning in linear time).

The learning of P&P grammars forms a hidden structure problem: every form of the language is generated by an underlying structure (the parameters) that cannot be directly observed from the data points. In fact, any single data point will contain many ambiguities as to which parameter setting could have generated it, as illustrated in (1).

(1) a. Data point: σ ˌσ σ ˈσ σ
b. Some compatible parameter settings:
Footing Direction: L-to-R; Foot Headedness: Right; Extrametricality: Off
(σ ˌσ)(σ ˈσ)(σ)
Footing Direction: R-to-L; Foot Headedness: Left; Extrametricality: Off
(σ)(ˌσ σ) (ˈσ σ)
Footing Direction: L-to-R; Foot Headedness: Left; Extrametricality: Left
<σ> (ˌσ σ) (ˈσ σ)

Since there is overwhelming evidence of statistical learning both in language (see, e.g., Saffran, Newport, & Aslin 1996) and beyond (see Kirkham et al. 2002 and many more), the most parsimonious theory would be that P&P grammars are learned by a domain-general statistical mechanism that does not have to be specified as part of UG. This is indeed what was proposed by Yang (2002) in his Naïve Parameter Learner (NPL), which is a maximally simple learning model that maintains a stochastic parameter grammar and updates it every time a data point comes in (i.e., it is an online learner).

However, Pearl (2011) argues that the NPL is insufficient to learn stress parameter settings: she shows that several variants of the NPL all consistently fail to acquire English stress from learning data representative of the input children receive. Instead, she argues (Pearl 2007, 2008, 2009a,b, 2011) that UG must contain both a P&P grammar and a set of domain-specific learning mechanisms. These domain-specific mechanisms refer to UG-specific information (i.e., the content of specific parameters) in order to solve the hidden structure puzzle created by parametric grammars.

Classic papers on P&P learning (see, e.g., Gibson and Wexler 1994, Berwick and Niyogi 1996, Lightfoot 1999) have proposed that parameters are learned with the help of cues. Dresher and Kaye (1990) and Pearl (2007) argue that, in addition to that, parameters must be ordered. Learning by cues means that every parameter comes with an innately specified configuration, a particular pattern in the data (a cue), that leads to the parameter being set one way or the other (see Dresher and Kaye 1990 for in-depth discussion of the nature of these cues). For example, the Extrametricality parameter may come with an associated cue that sets Extrametricality to 'off' if the learning data includes forms with stress on initial and on final syllables (see Dresher and Kaye 1990 and Pearl 2007, 2011 for a full overview of a possible cue systems for Dresher and Kaye's framework). Parameter ordering means that the learner only considers one parameter at a time in a particular order specified in UG; for instance, all evidence for Quantity (In)Sensitivity is considered before all evidence for Footing Directionality (Dresher and Kaye 1990). Both of these mechanisms differ fundamentally from standard learnability approaches for OT (e.g. Tesar 1995), where learning relies only on the basic architecture of the theory, without reference to the content of particular constraints or to patterns in the data.

An alternative model also set forth by Pearl (2007) replaces cues by Fodor's (1998) and Sakas and Fodor's (2001) parsing approach to ambiguity, which is a domain-general way of determining which parameter settings, if any, are essential for parsing a particular form. However, according to Pearl, the parsing approach would still need to be coupled with an innate parameter ordering, so that domain-specific learning mechanisms must be invoked regardless.

Here, we propose a novel domain-general statistical learner, the Expectation Driven Parameter Learner (EDPL), that we argue is a promising alternative approach for learning stress parameters, weakening the argument for domain-specific mechanisms. This learner maintains the crucial aspects of the NPL (stochastic parameter grammar, online gradual update mechanism), but slightly expands on the statistical mechanisms available to the learner, basing them on the well-established framework of Expectation Maximization (Dempster et al. 1977). These statistical mechanisms are still very efficient in terms of the time needed for their component computations, which is linear in the number of parameters.

We show that the EDPL's enhanced statistical inference abilities enable it to better cope with the ambiguity that is an inherent part of learning metrical stress. Given a representative test set of languages generated by Dresher and Kaye's (1990) system, we will show that, whereas the NPL has a success rate of only 4.3%, the EDPL has a success rate of 96%. This remarkable result suggests that what the NPL lacks is not domain-specific learning mechanisms (as argued by Pearl 2011), but more powerful statistical inference capabilities.

One alternative way to cope with ambiguity in P&P learning without domain-general mechanisms has been proposed by Gould (2015). This is a Bayesian approach that uses Dirichlet priors to remember how consistently each parameter setting has been rewarded at past learning trials. A similar, but distinct Bayesian approach was proposed by Pearl 2007, 2011, and failed at learning English stress, but perhaps Gould's solution is more effective. This solution is quite different from ours (crucially, it does not take separate samples from the grammar for each individual parameter setting). Given this difference, future work should compare both approaches on the same data sets to see if they produce similar effects.

The rest of this paper will be structured as follows. Section 2 will offer some necessary background on statistical learning of parameters; section 3 will then elaborate our own proposal (the EDPL), after which section 4 will describe our simulations and their results. Finally, section 5 will offer a general discussion of our findings as well as concluding remarks.

2 Statistical learning of parameters

Defining a finite number of parameter settings yields a finite, and oftentimes small, learning space. Indeed, an original argument for the P&P approach comes from learnability considerations: it is often assumed that reducing the space of possible hypotheses a learner could entertain simplifies the learning problem (see, e.g., Chomsky 1981/1982:3-4). For instance, Dresher and Kaye's (1990) set of 10 parameters of metrical structure yields $2^{10} = 1024$ different grammars (which is reduced further by Dresher and Kaye's stipulations on parameter value co-occurrence; their parameter 9, defooting in clash, is not taken into account, as they argue that it should be a part of a separate destressing/defooting module), and only 216 distinct languages. In theory, this means that a learner could arrive at the correct grammar for a data set by brute force enumeration of the possible grammars, checking each against the data.

However, a brute force approach is incompatible with evidence about the basic nature of human language learning, in particular the gradualness of learning. Longitudinal studies of language development have overwhelming found that acquisition is characterized by a gradual progression between closely related grammars (see, e.g., Fikkert 1994, Levelt et al. 2000, Demuth 1996), rather than categorical jumps between unrelated grammars as may be expected from brute force approach. Therefore, we favor a learning approach wherein learners systematically update their grammatical hypothesis, transitioning between closely related grammars until they have found one that minimizes error (or maximizes likelihood). One example of this kind of gradual approach for Optimality Theory is the well-known Gradual Learning Algorithm (Boersma 1997, Boersma and Hayes 2001). Brute force learning is also incompatible with a fundamental characteristic of phonological systems that any learning theory must ultimately face: variation and exceptionality. This is because the brute force approach presupposes absolute consistency between some target grammar in the finite set and all the learning data. Furthermore, stress parameters are a tiny part of the linguistic system children must ultimately acquire; it is highly unlikely that a brute-force approach could scale to the full language learning problem. For these reasons, we reject the brute-force approach and pursue a statistical learning model that has potential to address these various desiderata.

Within the P&P framework, such an approach is represented by Yang's (2002) NPL. Yang advocates this approach as a domain-general way to learn parameter grammars with a minimum of computation and inference. The NPL maintains a stochastic parameter grammar (not dissimilar to Stochastic OT, Boersma 1997) and updates this grammar every time a new data point is presented to the learner (just like for the Gradual Learning Algorithm, Boersma 1997) using the update rule in (3) below. We show in section 3 that the difference between the NPL and our own proposal, the EDPL, lies only in the way that the reward value in the update rule is computed, while the probabilistic grammar and the update rule are held constant.

A stochastic parameter grammar maintains a separate probability for each parameter over the possible settings it can take on. Analogously to Stochastic OT, these probabilities are used at production time to select categorical settings. This is done by independently flipping a weighted coin for each parameter to select its (categorical) setting. Once all parameters are set, the resulting combination of parameter settings can be used to generate the stress pattern. An example is shown in (2a) for two parameters: the foot headedness

(FootHead) and footing direction (Footing) parameters.[1] According to this grammar, feet are somewhat more likely to be trochaic (0.6) than iambic (0.4) and are more likely to be constructed right-to-left (0.7) than left-to-right (0.3). This gives rise to four logically possible scenarios, two of which are illustrated in (2b,c), along with the probabilities of choosing each scenario given the stochastic grammar G.

Both NPL and EDPL utilize this production process during learning, exploiting the learner's ability to use its current stochastic grammar to generate a stress pattern for a given learning datum and compare that stress pattern to the observed one. When the predicted stress pattern is compared to the attested stress pattern for that data point, a match or mismatch is recorded, as also shown in (2b,c). Match or mismatch is assessed solely based on the observable stress pattern: the footing that underlies the stress assignment is irrelevant.

(2) a. $G = \begin{Bmatrix} P(FootHead\text{: }L) = 0.6 & P(Footing\text{: }L \rightarrow R) = 0.3 \\ P(FootHead\text{: }R) = 0.4 \text{ ,} & P(Footing\text{: }R \rightarrow L) = 0.7 \text{ , ...} \end{Bmatrix}$

b. $P(\textit{FootHead: L, Footing: } R \rightarrow L \mid G) = 0.6 \times 0.7 = 0.42$
Data point: ˈka ta ˌpaa ˌta na
Predicted stress: (ˈka.ta)(ˌpaa)(ˌta.na) **Match**

c. $P(\textit{FootHead: R, Footing: } L \rightarrow R \mid G) = 0.4 \times 0.3 = 0.12$
Data point: ˈka ta ˌpaa ˌta na
Predicted stress: (kaˈta)(ˌpaa)(taˌna) **Mismatch**

Learning proceeds as follows. The learner starts with some initialization of the probabilistic grammar that is then updated online as data points are processed one-by-one. We chose to initialize with a uniform distribution for each parameter (each setting has 0.5 probability). This means the learner considers all settings of each parameter equally likely, an initial state that encodes no prior preferences about parameter settings. Other initializations, reflecting some prior tendencies in favor of certain parameter settings, may also be considered.

After a data point has been processed, the probabilistic grammar is updated according to the Linear Reward-Penalty Scheme (Bush and Mosteller 1951), which can be formulated as in (3). According to this formula, the probability of a parameter setting ψ_i at time $t + 1$ is a weighted sum of its probability at time t and its reward value $R(\psi_i)$. λ is the learning rate, a value between 0 and 1, regulating how close the updated probability should be to the current probability: the lower λ is, the more conservative the update. $R(\psi_i)$ is the reward value for parameter ψ_i – also between 0 and 1 – and is determined by processing the learning datum at time t. When $R(\psi_i) = 0$, the parameter setting's probability will decrease by an amount of $\lambda \times \hat{P}(\psi_i \mid G_t)$. When $R(\psi_i) = 1$, the parameter setting's probability will increase by an amount of $\lambda \times (1 - \hat{P}(\psi_i \mid G_t))$. Finally, when $R(\psi_i) \approx P(\psi_i \mid G_t)$, the parameter setting's probability will remain roughly the same.

(3) *Linear Reward-Penalty Scheme, used in both the NPL and the EDPL*
For each parameter setting ψ_i(FootHead: L, FootHead: R, etc.) and at each time (iteration) t:

$\hat{P}(\psi_i \mid G_{t+1}) = \lambda \times R(\psi_i) + (1 - \lambda) \times P(\psi_i \mid G_t)$ where

$P(\psi_i \mid G_t)$ is the parameter setting's probability in the grammar at time t

$R(\psi_i)$ is the parameter setting's current reward value, between 0 and 1 (see below)

λ is the learning rate, between 0 and 1; we chose $\lambda = 0.1$

The reward value $R(\psi_i)$ in the formula in (3) is computed differently for the NPL and the EDPL. The NPL method is discussed here, while the EDPL method will be introduced in section 3 below. The NPL computes $R(\psi_i)$ of a data point by producing its stress pattern once with the current grammar and assessing its match with the observed stress pattern. Let us call all parameter settings that were chosen during production, ψ_{chosen}. For all parameter settings in ψ_{chosen}, $R(\psi_i)$ is 1 when a match is recorded and 0 otherwise. $R(\psi_i)$ for parameter settings not in ψ_{chosen} equals 1 minus the $R(\psi_i)$ value for the parameter settings in ψ_{chosen}. For example, if trochees were used to generate the stress pattern, and the stress pattern was a match, then $R(\textit{FootHead: L})$

[1] Throughout this paper, we will refer to left-headed feet as trochees and to right-headed feet as iambs without implying the technical meaning of trochees and iambs in Hayes' (1995) framework.

would be 1, and $R(\textit{FootHead: R})$ would be 0. If these settings instead resulted in a mismatch, the reward values would be reversed.

Thus, the NPL will reward or penalize all parameter settings chosen for generating the current data point equally, depending on whether there happens to be a match or mismatch. In other words, the reward does not differentiate or encode the influence of each individual parameter setting on producing a match or mismatch. This minimizes the model's explicit attempts to deal with the so-called Credit Problem (Dresher 1999): because of the ambiguity between parameter settings for every data point (as illustrated in (1)), it is unclear which parameter settings to blame for a mismatch, and which ones to give credit for a match. The NPL does not attempt to disambiguate: all parameter settings are assigned equal credit in case of a match and equal blame in case of a mismatch.

There are two problematic scenarios that can arise due to the Credit Problem, pointed out by Yang (2002). In the first scenario, shown in (4), a parameter setting essential to generating the observed stress pattern, Main Stress: Left (Main: L), is nevertheless penalized as an "accomplice" in case of a mismatch because incompatible parameter settings, Extrametricality: On & Left (Ext: L), were also chosen during production. As shown in (4c), both parameter settings receive reward values of 0 in case of a mismatch.

(4) *The accomplice scenario, NPL*
   a. Data point: ˈka ta ˌpaa ˌta na
      Necessary setting – Main Stress: Left
      Incompatible setting – Extrametricality: Left
   b. Predicted stress: <ka> (ˈta)(ˌpaa)(ˌta na) **Mismatch**
   c. $R(\textit{Main: L}) = R(\textit{Ext: L}) = 0$

In the second scenario, shown in (5), a parameter setting irrelevant to an observed stress pattern is rewarded as a "hitchhiker" in the case of a match because all essential parameter settings responsible for producing the correct pattern were selected. In this case, CVC Syllables: Heavy is irrelevant since there are no CVC syllables in this data point, while Main Stress: Left is essential and responsible for producing the matching pattern. The NPL does not differentiate these two parameter settings, assigning each a reward of 1, as shown in (5c).

(5) *The hitchhiker scenario, NPL*
   a. Data point: ˈka ta ˌpaa ˌta na
      Necessary setting – Main Stress: Left
      Irrelevant setting – CVC Syllables: Heavy
   b. Predicted stress: (ˈka ta)(ˌpaa)(ˌta na) **Match**
   c. $R(\textit{CVC: Heavy}) = R(\textit{Main: L}) = 1$

NPL has no mechanism for differentiating essential settings from hitchhikers and accomplices. In the next section, we will show how the alternative reward value assignment in the EDPL solves these two problems by making reward values sensitive to the degree of necessity or incompatibility between a parameter setting and the desired stress pattern.

3 The Expectation Driven Parameter Learner (EDPL)

The EDPL model proposed here extends Jarosz's (2015) Expectation Driven Learning proposal developed for probabilistic OT. It is identical to the NPL in that it maintains the same type of stochastic grammar (see (2)), starts with a uniform distribution over parameter settings, and uses the same update rule (3) after each data point presented to the learner. The difference between the two models lies in their computation of the reward value. In a nutshell, the EDPL sets the reward value $R(\psi_i)$ for parameter setting ψ_i equal to the probability of that parameter setting given the current data point and the current grammar G_t: $R(\psi_i) = \hat{P}(\psi_i | \textit{ data point, } G_t)$. These computations are an online (non-batch) approximation of Expectation Maximization (EM; Dempster et al. 1977): instead of setting the new probabilities of parameter settings to their expected values given the entire data set, which would be a classic EM approach, the process here is broken up into smaller and more local calculations computed incrementally on each incoming data point, just like in the NPL.

The computation of $R(\psi_i)$ relies on two crucial steps, as proposed by Jarosz (2015): estimating the probability of the current data point given a particular parameter setting – $\hat{P}(\textit{data point} \mid \psi_i, G_t)$ – with the help of sampling, and converting this probability into $P(\psi_i \mid \textit{data point}, G_t)$ using Bayes' Rule.

To estimate the term $\hat{P}(\textit{data point} \mid \psi_i, G_t)$ the current parameter grammar, G_t, is temporarily modified by replacing the probability of parameter setting ψ_i with 1 (and replacing the probability of all other settings of the same parameter with 0). This temporary grammar in which every production is guaranteed to choose parameter setting ψ_i can provide an estimate of the influence that ψ_i has on matching the current data point. For this estimate, a number of samples are taken (the number of samples is represented by r; we chose $r = 50$) by repeatedly producing the current data point with that grammar, and assessing match and mismatch as in (2b,c) in section 2. An estimate of $P(\textit{data point} \mid \psi_i, G_t)$ is then obtained by dividing the number of matches by the sample size, as shown in (6a). In other words, the proportion of production samples using parameter ψ_i that lead to matches are used to estimate the probability of generating the data point given ψ_i. The analogous computation is then performed for the other setting of that parameter: $\neg\psi_i$. During this process, the probabilities for the other parameters are left as is. This makes it possible to directly compare the ability of ψ_i and $\neg\psi_i$ to generate the data point, effectively isolating the effects of manipulating this particular parameter.

Once we have an estimate of $\hat{P}(\textit{data point} \mid \psi_i, G_t)$ for all settings of a parameter, we can use Bayes' Rule to restate this conditional probability in terms of $\hat{P}(\psi_i \mid \textit{data point}, G_t)$, as illustrated in (6b). The term $P(\psi_i \mid G_t)$ stands for the parameter setting's probability in the current grammar, which only needs to be looked up. The term $\hat{P}(\textit{data point} \mid G_t)$ is the weighted sum of $\hat{P}(\textit{data point} \mid \psi_i, G_t)$ for all settings of the current parameter; in the equation, $\neg\psi_i$ stands for the opposite setting of ψ_i (for instance, if $\psi_i = \textit{Main: Left}$, then $\neg\psi_i = \textit{Main: Right}$).

(6) *Reward value computation for the EDPL*

a. $$\hat{P}(data\ point|\psi_i, G_t) = \frac{number\ of\ matches\ given\ G_t\ with\ P(\psi_i)=1}{r}$$

b. $$R(\psi_i) \equiv \hat{P}(\psi_i|data\ point, G_t) = \frac{\hat{P}(data\ point|\psi_i, G_t) \times P(\psi_i|G_t)}{\hat{P}(data\ point|G_t)}$$

c. $$\hat{P}(data\ point|G_t) = \hat{P}(data\ point|\psi_i, G_t) \times P(\psi_i|G_t)\ + \hat{P}(data\ point|\neg\psi_i, G_t) \times P(\neg\psi_i|G_t)$$

While this involves a little more computation than the NPL, this computation is still highly efficient and grows slowly (linearly) in the number of parameters. For the NPL, the number of times the grammar is called for the reward value computation is a constant 1. For the EDPL, this number equals the number of parameters × 2 (settings for each parameter) × r (= 50), meaning 100 times the number of parameters in our simulations.

The EDPL reward value computation makes it possible to make considerable progress on the Credit Problem. The problematic scenarios in (4, 5) are resolved by the fact that reward values in EDPL are both specific to each parameter setting and sensitive to the parameter setting's relevance to the data point.

In the "accomplice" scenario (repeated in (7a), from (4a)), the EDPL distinguishes between the necessary parameter setting Main Stress: Left and the incompatible setting Extrametricality: Left because the two parameter settings' reward values are computed independently (see (6a)). Since Main Stress: Right and Extrametricality: Left are both incompatible with initial main stress in ['ka ta ˌpaa ˌta na], these settings will inevitably yield zero matches, so that $\hat{P}(\textit{data point} \mid \textit{Main: R}, G_t)$ and $\hat{P}(\textit{data point} \mid \textit{Ext: L}, G_t)$ will be 0. For $R(\textit{Ext: L})$, shown in (7b), this means that the numerator in the formula in (6b) equals 0, yielding an outcome of 0 as long as $\hat{P}(\textit{data point} \mid \textit{Ext: R}, G_t) \times \hat{P}(\textit{Ext: R} \mid G_t) > 0$. When computing $R(\textit{Main: L})$, shown in (7c), plugging in 0 for $P(\textit{data point} \mid \textit{Main: R}, G_t)$ in the formula in (6c) yields an outcome of 1 as long as $\hat{P}(\textit{data point} \mid \textit{Main: L}, G_t) \times \hat{P}(\textit{Main: L} \mid G_t) > 0$. In other words, essential parameter settings will generally yield reward values of 1, and incompatible settings will generally lead to reward values of 0. Essential parameter settings are not penalized for accomplices' behavior because that behavior is constant across the relevant comparisons. The consequences of the essential parameter setting and its antagonist (other setting of same parameter) are only compared to one another, effectively factoring out the independent effects of any potential accomplices' behavior.

(7) *No accomplice effect, EDPL* (cf. (4))

a. Data point: ˈka ta ˌpaa ˌta na
Necessary setting – Main Stress: Left
Incompatible setting – Extrametricality: Left

b. $R(Ext{:}\,L) = \frac{0}{0+\hat{P}(\text{'ka ta ˌpaa ˌta na}|Ext{:}R,G_t) \times P(Ext{:}R|G_t)} = 0$

c. $R(Main{:}\,L) = \frac{\hat{P}(\text{'ka ta ˌpaa ˌta na}|Main{:}L,G_t) \times P(Main{:}L|G_t)}{\hat{P}(\text{'ka ta ˌpaa ˌta na}|Main{:}L,G_t) \times P(Main{:}L|G_t)+0} = 1$

In the situation where the NPL rewards an irrelevant parameter setting as a "hitchhiker" (see (8a), repeated from (5a)), the EDPL once again gives Main Stress: Left the maximal reward value of 1, but because the reward of CVC Syllables: Heavy (CVC: Heavy) is assessed independently, this parameter setting is not rewarded with equal strength. Rather, CVC: Heavy and CVC: Light should yield an approximately equal amount of matches in the sampling procedure, so that $\hat{P}(\textit{data point} \mid \textit{CVC: Heavy, } G_t) \approx \hat{P}(\textit{data point} \mid \textit{CVC: Light, } G_t)$. This means that we can practically swap out $\hat{P}(\textit{data point} \mid \textit{CVC: Heavy, } G_t)$ for $\hat{P}(\textit{data point} \mid \textit{CVC: Light, } G_t)$ in the formula in (6b). If we perform this swap, we can simplify this expression to just $\hat{P}(\textit{CVC: Heavy} \mid G_t)$, as shown in (8b). As above, the model is able to detect the irrelevance of the parameter because the comparison is limited to the settings of that parameter, while holding everything else constant, and an irrelevant parameter will produce the same outcome regardless of its setting.

(8) *No hitchhiker effect, EDPL* (cf. (5))

a. Data point: ˈka ta ˌpaa ˌta na
Necessary setting – Main Stress: Left
Irrelevant setting – CVC Syllables: Heavy

b. $R(CVC{:}\,Heavy) \approx \frac{\hat{P}(\text{'k...}|CVC{:}Heavy,G_t) \times P(CVC{:}Heavy|G_t)}{\hat{P}(\text{'k...}|CVC{:}Heavy,G_t) \times \left(\hat{P}(CVC{:}\,Heavy|G_t) + P(CVC{:}\,Light|G_t)\right)}$

$= \frac{\hat{P}(\text{'k ...}\,|CVC{:}\,Heavy,\,G_t) \times P(CVC{:}\,Heavy|G_t)}{\hat{P}(\text{'k ...}\,|CVC{:}\,Heavy,\,G_t) \times 1} = \hat{P}(CVC{:}\,Heavy|G_t)$

In sum, the necessary setting Main Stress: Left in (7) is given the maximal reward value of 1, since its reward value is assessed independently of the incompatible setting Extrametricality: Left. In (8), the irrelevant setting CVC Syllables: Heavy is neither rewarded nor penalized – as can be verified with the formula in (3), a reward value (roughly) equal to the old probability of the parameter value yields no change. Both of these solutions are possible because the updates are computed independently for each parameter, after individually assessing the impact of each setting of that parameter on the grammar's ability to generate the observed stress pattern.

The EDPL also makes it possible to exploit conditional dependencies between hypotheses to let prior knowledge influence learning. Consider (9): there is a dependency between Extrametricality and Foot Headedness - matches are possible only if both are set to Left or both are set to Right - but it is not clear from this data point alone which hypothesis is to be chosen. However, if the learner has already determined that Extrametricality: Left should be preferred over Extrametricality: Right, then this data point should provide support for Foot Headedness: Left, since trochaic feet are the only way to produce a match given left extrametricality. The calculation of the reward value reflects this intuition. Since Extrametricality: Left is more likely, Foot Headedness: Left will be more successful in producing matches than Foot Headedness: Right when the Foot Headedness parameter is tested. This, in turn, means that Foot Headedness: Left will be rewarded more than Foot Headedness: Right; that is, trochaic feet will be rewarded. Conversely, if right extrametricality were preferred by previous data points, the learner would reward iambic feet based on this data point. Thus, these updates derive inferential behavior that is most consistent with the current data points and prior beliefs, allowing the learner to make stronger inferences during learning. While the data point in isolation is ambiguous, the data point combined with the learner's prior knowledge makes it possible to make headway on the learning problem despite rampant ambiguities in the data.

(9) *Conditional dependencies*

Data point: ma ˌna ma ˈna na		
Ext: L and FootHead: L	<ma> (ˌna ma) (ˈna na)	**Match**
Ext: L and FootHead: R	*<ma> (na ˌma) (na ˈna)	**Mismatch**
Ext: R and FootHead: L	*(ˌma na) (ˈma na) <na>	**Mismatch**
Ext: R and FootHead: R	(ma ˌna) (ma ˈna) <na>	**Match**

Summarizing, the EDPL computes a separate reward value for each parameter, and this reward value is equal to the probability of that parameter given the current data point and grammar. Calculating the reward value approximates the E step of the EM algorithm, wherein the learner's current grammatical hypothesis is used to calculate expected values of hidden variables (parameter settings, in this case) for the observed data. The update rule then approximates the M step, making the grammar more compatible with those expected hidden variables. Following Jarosz (2015), the algorithm uses sampling and Bayes' Rule to estimate this probability. This increased flexibility and sensitivity provides a solution to Credit Problem, as shown in examples (7-9) above.

4 Tests of the NPL and EDPL

To evaluate the EDPL model and compare it to the NPL, we tested both the NPL and the EDPL on a data set that systematically explores the stress parameter set proposed by Dresher and Kaye (1990). To our knowledge, this is the first typologically extensive test of the NPL. We chose Dresher and Kaye's stress parameter set because this system has been explored in several prior learning studies: both by Dresher and Kaye themselves, and by Pearl (2007, 2008, 2011).

Our data set explored all interactions between foot properties, quantity-sensitivity, and extrametricality – which was achieved by varying the settings of the 6 parameters shown in (10a). The 4 parameter settings shown in (10b) were held constant in the data set. The learner itself did not know about this lack of variation on the parameters in (10b), and it had to find the settings of all 10 parameters without prior knowledge.

(10) a. *Parameters whose settings were varied*
- Foot Headedness: Left/Right
- Foot Boundedness: On/Off
- Deletion of Light-syllable-headed Feet: On/Off
- Quantity Sensitivity: On/Off
- Extrametricality: On/Off
- Extrametricality: Left/Right

b. *Parameter settings held constant*
- Main Stress: Left
- Footing Direction: L → R
- Secondary Stress: Spelled Out
- CVC Syllables: Heavy

The six parameters in (10a) form the core of the hidden structure learning problem: extrametricality and foot location and headedness trade off in explaining the location of stress (as illustrated in (9) above), and the potential for quantity-sensitivity and deletion of light-syllable-headed feet add further ambiguity as to the structure underlying the stress pattern.

As can be seen in (10), Dresher and Kaye's parameter set is completely symmetrical: in contrast to Hayes' (1995) system, there is no ban on (Heavy Light) feet or quantity-insensitive iambs. As illustrated in (11), languages that assign 'mirror image' stress patterns have the same interaction between parameters: the only difference between them is that all parameter settings that have to do with left and right edges are flipped. Therefore, mirror image languages face the same learning challenges. In order to keep our data set compact, we avoided 'mirror images' by keeping the main stress foot edge and the footing direction constant. Secondary stress assignment and the weight of CVC syllables were kept constant because they are learning challenges that do not interact with the position of foot heads and edges.

(11) *Mirror image languages*

Language A	Language B, mirror image of Language A
(ˈσ σ)(ˌσ σ) <σ>	<σ> (σ ˌσ) (σ ˈσ)
(ˈσ σ) (ˌσ) <σ>	<σ> (ˌσ) (σ ˈσ)
Main: **L**, FootHead: **L**, FootDir: **L → R**	Main: **R**, FootHead: **R**, FootDir: **R → L**
Extrametricality: On and **Right**	Extrametricality: On and **Left**
Foot Boundedness: On, QS: Off	Foot Boundedness: On, QS: Off
Deletion of Light-syllable-headed Feet: Off	Deletion of Light-syllable-headed Feet: Off
Secondary Stress: Spelled Out	Secondary Stress: Spelled Out

The six varying parameters defined $2^6 = 64$ parameter setting combinations, which yielded 23 distinct languages: 8 quantity-insensitive languages, and 15 quantity-sensitive ones. Although 16 of these correspond to an attested stress pattern in StressTyp2 (Goedemans et al. 2015) modulo secondary stress deletion and/or mirror image creation, attested cases were not found for all of the 23 languages. To expose the learner to all 23 languages and make the presentation of each language to the learner maximally uniform, we presented each language's stress pattern on a set of constructed words – namely, on all 3- to 6-syllable combinations of the syllables [ta], [taa], and [tan]. Each word was presented to the learner with equal probability, and there were no exceptional words.

Pearl (2011) tested the NPL on a completely different data set – namely, on a corpus of child directed speech in American English. Since English has a sizeable number of exceptional words (Pater 2000 and references therein), and the stochastic parameter grammar has no way of representing exceptions, this could have led to additional difficulties for the NPL. Our data set contains no exceptional words and therefore eliminates this confound in testing the NPL's ability to learn stress parameter settings.

Both the NPL and the EDPL were implemented in R (R core team 2013), and tested on each of these 23 languages. For each language and each learner, 10 runs were performed, and each run continued for 1,000,000 iterations or until successful convergence, which was assessed every 100 iterations. We defined a run as successfully convergent when the resulting grammar produced each word's stress pattern correctly 99 out of 100 times. A learner's success on our data set was measured by both the rate of successful convergence and the number of iterations needed to reach successful convergence.

The results, summarized in (12), were quite dramatic, as already anticipated in section 1: the NPL showed a learning rate of only 4.3%, while the EDPL learned the same data set 96% of the time, and in the cases that the NPL did converge, it was over 400 times slower on average than the EDPL.

(12) *Results for the NPL and the EDPL*

	NPL	EDPL
Successful convergence	Occurred in 1/23 languages (10/10 runs for 1/23 languages, 0/10 runs for 22/23 languages) Overall: **4.3%**	Occurred in 23/23 languages (10/10 runs for 22/23 languages, 1/10 runs for 1/23 languages) Overall: **96%**
Iterations untill successful convergence (if reached)	Average: 89,370 (Range: 16,300-204,600)	Average: 203 (Range: 100-400)

Thus, we can confirm Pearl's (2011) finding that the NPL fails on stress parameter setting, and, moreover, we can confirm that this is not caused by the English data set (most notably, its wide range of exceptionality). The only language that the NPL did consistently learn is a language with only leftmost main stress and no other stress phenomena. Out of the 23 languages considered, this language is consistent with the largest number of different parameter settings, which makes it is the easiest to arrive at by random guessing. The fact that the NPL found several wildly divergent parameter settings for this language is evidence for this. Strikingly, this language was not the fastest to be learned for the EDPL – its successful convergence time was 210 iterations on average, while the fastest languages (quantity-insensitive L-to-R iambs with left or right extrametricality) were both learned within 100 iterations (successful convergence was tested every 100 iterations). Recall that the update rule was identical for the two learners, and the same learning rate was used. This dramatic difference in learning speed therefore indicates that the EDPL's updates were more effective.

At the same time, the EDPL showed a high success rate. The only language for which the EDPL did not reach successful convergence on all runs is a quantity-insensitive L-to-R trochaic language with left

extrametricality, as shown in (13a). The special property of this language is that all of its forms allow both a trochaic and an iambic analysis. Only a trochaic analysis is compatible with all forms, but it requires finding that extrametricality is on and at the left; an iambic analysis can yield matches (shown in gray) when extrametricality is either off or on the right, as can be seen in (13b,c). Because the iambic analysis is compatible with a broader set of possibilities, the learner tends to initially favor this analysis, while gradually lowering the probability of left extrametricality. Once the probability of left extrametricality is sufficiently low, the learner has fallen for the trap set by this data set, and will never be able to recover.

(13) *The only language for which the EDPL did not always converge, and its distractor languages*

QI L-to-R languages	3 syllables	4 syllables	5 syllables	6 syllables
a. Trochees, Ext. L	<σ> (ˈσ σ)	<σ> (ˈσ σ)(ˌσ)	<σ> (ˈσ σ)(ˌσ σ)	<σ> (ˈσ σ)(ˌσ σ)(ˌσ)
b. Iambs, Ext. R	(σ ˈσ) <σ>	(σ ˈσ)(ˌσ) <σ>	(σ ˈσ)(σ ˌσ) <σ>	(σ ˈσ)(σ ˌσ)(ˌσ) <σ>
c. Iambs, No Ext.	(σ ˈσ)(ˌσ)	(σ ˈσ)(σ ˌσ)	(σ ˈσ)(σ ˌσ)(ˌσ)	(σ ˈσ)(σ ˌσ)(σ ˌσ)

It is unclear to which degree the difficulty that the EDPL has with this language is problematic. While the pattern in (13a) is predicted by Dresher and Kaye's (1990) model, evidence for Trochees with left extrametricality is rather scarce (see Hyde 2011 for arguments against purported cases of left extrametricality; however, see Melinger's 2002 analysis of Seneca pitch accent for a potential example of left extrametricality). Our implementation excludes a L-to-R iambic parse with clash avoidance, which would be a more straightforward way to analyze the language in (13a). A more comprehensive implementation including clash avoidance (see Dresher and Kaye 1990 and Dresher 1999 for a description of a destressing module that would be compatible with the parameter set we used) may allow the learner a way out of the ambiguity described in the paragraph above example (13). In any case, the EDPL was able to find a trochaic solution for this language at least once, and it found solutions for all languages in the typology in general, whereas the NPL was only able to learn one of the languages.

5 General discussion

Parametric models of grammar remain relevant in recent literature (Pearl 2007, 2008, 2009a,b, 2011, Sakas and Fodor 2012, Gould 2015). Whereas domain-general statistical learning mechanisms for learning stress in constraint-based grammars has been explored extensively for Optimality Theory and related approaches (Tesar and Smolensky 2000, Boersma and Pater 2016, Jarosz 2015), the learning of parameter-based grammars for word stress was found to be resistant to similar domain-general mechanisms (in particular, Yang's (2002) Naïve Parameter Learner (NPL)) in previous work (Pearl 2007, 2011). Pearl argues that learners must employ domain-specific mechanisms (parameter ordering and cues, specified in UG – see also Dresher and Kaye 1990). In this paper, we propose a novel statistical learner for parametric grammars: the Expectation Driven Parameter Learner (EDPL), and show that this learner exhibits strong performance (>95% successful convergence) on a typologically extensive dataset, making the case for the viability of an approach without domain-specific learning mechanisms.

Our EDPL proposal retains the advantages of the NPL: probabilistic parameter grammars are learned gradually with a very simple update rule, and computation is linear in the number of parameters. However, the EDPL has stronger inferential abilities than the NPL, which makes it better equipped to distinguish between necessary, incompatible, and irrelevant parameters for a given data point (a challenge known as the Credit Problem; Dresher 1999). As shown in section 3, this is because the EDPL computes the value by which parameter settings' probabilities are updated (the reward value) for each parameter setting individually, and this value is influenced by both the relative success of a parameter setting on a data point and the information already contained in the current grammar.

As can be seen in section 4, testing the NPL and the EDPL on 23 languages that systematically explore Dresher and Kaye's (1990) parameters set yields a formidable advantage for the EDPL: the NPL is successful in 4.3% of all trials, while the EDPL is successful in 96% of all trials. This, to our knowledge, is the first typologically extensive test of the NPL. This confirms Pearl's (2011) finding that the NPL is highly ineffective for stress parameters, and confirms that the potential confound of lexical exceptions in Pearl's English data set is not to blame for the NPL's failure, since our data set contained no lexical exceptions.

The fact that the EDPL has a very high rate of success in learning stress in Dresher and Kaye's (1990) parameter framework severely weakens Pearl's (2007, 2011) interpretation of the NPL's failure at learning

stress parameters. Pearl argues that domain-specific mechanisms are necessary to remedy the failure of statistical, domain-general learners such as the NPL. However, as summarized above, our statistical, domain-general learner – the EDPL – is able to learn stress parameters by alleviating the statistical shortcomings of the NPL approach. This suggests that the failure of the NPL does not lie in its lack of domain-specific mechanisms, but in its lack of a sufficiently developed statistical inference mechanism.

The NPL's main shortcomings consist of rewarding all parameter settings equally and rewarding them categorically, which leads to sometimes penalizing parameter settings necessary for a data point's production or rewarding parameter settings irrelevant to a data point's production (see section 2). The EDPL instead rewards parameter settings in proportion to their contribution to increasing the accuracy of the grammar on the current data point, by invoking an online version of Expectation Maximization (Dempster et al. 1977) along the lines of the learner proposed in Jarosz (2015), as described in section 3.

If it is indeed the case that statistical learning is sufficient to learn stress parameters, as preliminarily confirmed by our findings, and given that statistical learning is also generally found sufficient to learn stress in Optimality Theory (Tesar 1995, Tesar and Smolensky 2000, Boersma 1997, Boersma and Hayes 2001, Boersma and Pater 2016, Jarosz 2013, 2015), we are able to argue that UG does not need to include a grammar model as well as a learning model (as suggested by Pearl 2007, 2011). Instead, it is plausible that UG only contains a grammar model, while the learning model is provided by general cognition, given that statistical learning is observed outside the realm of language as well (see, e.g., Kirkham et al. 2002). A UG that only contains a grammar model and not a learning model, *ceteris paribus*, would increase parsimony.

In future work, we hope to apply this model to a wider range of parametric grammar learning problems in order to corroborate the result obtained here. In addition, it will be important to incorporate a clash avoidance module as described by Dresher and Kaye (1990) to test our conjecture that the language problematic for the EDPL may be learned more easily with such a module (see section 4 for this prediction). Moreover, in order to further strengthen our argument, we hope to provide explicit simulations of learning the same data set with Pearl's (2007, 2008) Bayesian learner with domain-specific mechanisms. Another statistical parameter learning model that enhances the NPL's power of inference has been developed by Gould (2015); comparing Gould's model with our own would also be an interesting next step.

Finally, incorporating the possibility to represent underlying stress marks (see Dresher 2016 for a proposal of how to do this for cue-based learning in parametric grammars) as well as lexical exceptions to the language-wide pattern would be a great asset for any model of learning parameters. Such extensions have been implemented in Jarosz's (2015) Expectation Driven Learning framework, upon which the EDPL is based (see Jarosz 2015 and Nazarov 2016: chapter 4). Using these techniques, and applying the learner to more intricate cases of parameter learning, we hope to be able to further support the hypothesis that statistical learning is sufficient for learning phonological grammar.

References

Archangeli, Diana, and Douglas Pulleyblank. 1994. *Grounded Phonology*. Cambridge, MA: MIT Press.

Berwick, Robert C., and Partha Niyogi. 1996. "Learning from Triggers." *Linguistic Inquiry* 27(4).605-622.

Blevins, Juliette. 1995. The syllable in phonological theory. *The handbook of phonological theory*, ed. by John Goldsmith, 206-44. Oxford: Basil Blackwell.

Boersma, Paul. 1997. "How we learn variation, optionality, and probability." *IFA Proceedings* 21.43-58.

Boersma, Paul, and Bruce Hayes. 2001. "Empirical tests of the Gradual Learning Algorithm." *Linguistic Inquiry* 32.45-86.

Boersma, Paul, and Joe Pater. 2016. "Convergence Properties of a Gradual Learning Algorithm for Harmonic Grammar." *Harmonic Grammar and Harmonic Serialism*, edited by John J. McCarthy and Joe Pater. London: Equinox Press.

Bush, Robert, and Frederick Mosteller. 1951. "A mathematical model for simple learning." *Psychological Review* 58.313–323.

Cho, Young-mee Yu. 1999. *Parameters of Consonantal Assimilation*. München: LINCOM Europa.

Chomsky, Noam. 1981/1982. *Lectures on government and binding*. Second revised edition. Dordrecht: Foris.

Dempster, A. P.; N. M. Laird; and D. B. Rubin. 1977. "Maximum Likelihood from Incomplete Data via the EM Algorithm." *Journal of the Royal Statistical Society. Series B (Methodological)* 39(1).1–38.

Demuth, Katherine. 1996. "The prosodic structure of early words." *From Signal to Syntax: Bootstrapping from Speech to Grammar in Early Acquisition*, ed. by James Morgan and Katherine Demuth, 171-84. Hillsdale, NJ: Lawrence Erlbaum Associates.

Dresher, B. Elan. 1999. "Charting the Learning Path: Cues to Parameter Setting." *Linguistic Inquiry* 30.27-67.

Dresher, B. Elan. 2016. "Covert representations, contrast, and the acquisition of lexical accent.", *Dimensions of phonological stress*, ed. by Jeffrey Heinz, Rob Goedemans, and Harry van der Hulst, 231–62. Cambridge: Cambridge University Press.

Dresher, B. Elan, and Jonathan D. Kaye. 1990. "A computational learning model for metrical phonology." *Cognition* 34.137-195.

Fikkert, Paula. 1994. *On the acquisition of prosodic structure*. Dordrecht: Holland Institute of Generative Linguistics.

Fodor, Janet D. 1998. "Parsing to Learn." *Journal of Psycholinguistic Research* 27(3).339-374.

Gibson, Edward, and Kenneth Wexler. 1994. "Triggers." *Linguistic Inquiry* 25(3): 407-454.

Goedemans, Rob W.; Jeffrey Heinz; and Harry van der Hulst. 2015. *StressTyp2, version 1*. Web download archive. Online: http://st2.ullet.net.

Gould, Isaac. 2015. *Syntactic Learning from Ambiguous Evidence: Errors and End-States*. Cambridge, MA: MIT dissertation.

Halle, Morris, and Jean-Roger Vergnaud. 1987. *An Essay on Stress*. Cambridge, MA: MIT Press.

Heinz, Jeffrey. 2011a. "Computational Phonology Part I: Foundations." *Language and Linguistics Compass* 5(4).140-52.

Heinz, Jeffrey. 2011b. "Computational Phonology Part II: Grammars, Learning, and the Future." *Language and Linguistics Compass* 5(4).153-168.

Hyde, Brett. 2011. "Extrametricality and Nonfinality." *The Blackwell Companion in to Phonology*, ed. by Marc van Oostendorp, Colin J. Ewen, Beth Hume, and Keren Rice, 1027-1051. Oxford, UK: Blackwell.

Jarosz, Gaja. 2013. "Learning with Hidden Structure in Optimality Theory and Harmonic Grammar: Beyond Robust Interpretive Parsing." *Phonology* 30(1).27–71.

Jarosz, Gaja. 2015. *Expectation Driven Learning of Phonology*. Ms., University of Massachusetts Amherst.

Kirkham, Natasha Z., Jonathan A. Slemmer, Scott P. Johnson. 2002. "Visual statistical learning in infancy: evidence of a domain general learning mechanism." *Cognition* 83.B35-42.

Kraska-Szlenk, Iwona. 1995. *The Phonology of Stress in Polish*. Urbana-Champaign, IL: University of Illinois, Urbana-Champaign dissertation.

Levelt, Clara C.; Niels O. Schiller; and Willem J. Levelt. 2000. "The acquisition of syllable types." *Language Acquisition* 8.237–64.

Lightfoot, David. 1999. *The Development of Language: Acquisition, Change, and Evolution*. Oxford: Blackwell.

Melinger, Alissa. 2002. "Foot Structure and Accent in Seneca." *International Journal of American Linguistics* 68(3).287-315.

Nazarov, Aleksei. 2016. *Extending Hidden Structure: Features, Opacity, and Exceptions.* Amherst, MA: University of Massachusetts Amherst dissertation.

Pater, Joe. 2000. "Non-Uniformity in English Secondary Stress: The Role of Ranked and Lexically Specific Constraints." *Phonology* 17(2).237–74.

Pearl, Lisa. 2007. *Necessary Bias in Natural Langauge Learning*. College Park, MD: University of Maryland dissertation.

Pearl, Lisa. 2008. "Putting the Emphasis on Unambiguous: The Feasibility of Data Filtering for Learning English Metrical Phonology." *Proceedings of the 32nd annual Boston University Conference on Child Language Development (BUCLD 32)*, ed. by Harvey Chan, Heather Jacob, and Enkeleida Kapia, 390–401. Somerville, MA: Cascadilla Press.

Pearl, Lisa. 2009a. "Learning English Metrical Phonology: When Probability Distributions Are Not Enough." *Proceedings of the 3rd Conference on Generative Approaches to Language Acquisition North America (GALANA 2008)*, ed. by Jean Crawford, Koichi Otaki, and Masahiko Takahashi, 200-11. Somerville, MA: Cascadilla Press.

Pearl, Lisa. 2009b. *Acquiring Complex Linguistic Systems from Natural Language Data: What Selective Learning Biases Can Do*. Ms., University of California, Irvine.

Pearl, Lisa. 2011. "When Unbiased Probabilistic Learning is Not Enough: Acquiring a Parametric System of Metrical Phonology." *Language Acquisition* 18(2).87-120.

Prince, Alan, and Paul Smolensky. 1993/2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Malden, MA: Blackwell.

R Core Team. 2013. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Online: http://www.R-project.org/.

Saffran, Jenny R.; Richard N. Aslin; and Elissa L. Newport. 1996. "Statistical Learning by 8-Month-Old Infants." *Science* 274(5294).1926-8.

Sakas, William G., and Janet D. Fodor. 2001. "The Structural Triggers learner." *Language Acquisition and Learnability*, ed. by Stefano Bertolo, 172-233. Cambridge, UK: Cambridge University Press.

Sakas, William G., and Janet D. Fodor. 2012. "Disambiguating Syntactic Triggers." *Language Acquisition* 19(2).83-143.

Tesar, Bruce B. 1995. "Computational Optimality Theory." Boulder, CO: University of Colorado Boulder dissertation.

Tesar, Bruce B., and Paul Smolensky. 2000. *Learnability in Optimality Theory*. Cambridge, MA: MIT Press.

Yang, Charles. 2002. *Knowledge and Learning in Natural Language*. Oxford: Oxford University Press.