On evaluation metrics in Optimality Theory[0]

Ezer Rasin and Roni Katzir

March 12, 2015

Abstract

We develop an evaluation metric for Optimality Theory that allows a learner to induce a lexicon and a phonological grammar from unanalyzed surface forms. We wish to model aspects of knowledge such as the English-speaking child's knowledge that the aspiration of the first segment of *k^{h}æt* is predictable and the French-speaking child's knowledge that the final *l* of *table* 'table' is optional and can be deleted while that of *parle* 'speak' cannot. We will try to show that the learner we present succeeds in obtaining this kind of knowledge and is better equipped to do so than other existing learners in the literature.

Keywords: learning, evaluation metrics, minimum description length, optimality theory, phonology

1 Introduction

We develop an evaluation metric for Optimality Theory (OT; Prince and Smolensky, 1993) that allows a learner to induce a lexicon and a phonological grammar from unanalyzed surface forms. We wish to model aspects of knowledge such as the English-speaking child's knowledge that the aspiration of the first segment of *k^{h}æt* is predictable and the French-speaking child's knowledge that the final *l* of *table* 'table' is optional and can be deleted while that of *parle* 'speak' cannot (Dell, 1981). We take it that any theory of phonology would require this knowledge to be learned rather than innate. We will try to show that the learner we present succeeds in obtaining this kind of knowledge and is better equipped to do so than other existing learners in the literature.

We start, in section 2, by constructing the evaluation metric, based on the principle of Minimum Description Length (MDL), a criterion growing out of a line of work pioneered by Solomonoff (1964) and used for various aspects of natural language by Berwick (1982), Rissanen and Ristad (1994), Stolcke (1994), Brent and Cartwright (1996), Grünwald (1996), de Marcken (1996), Clark (2001), Goldsmith (2001, 2010), Dowman (2007), Chater and Vitányi (2007), Hsu and Chater (2010), Hsu et al. (2011), and Goldsmith and Riggle (2012), among others; see also the very closely related Bayesian approach to grammar induction, used by Solomonoff (1964), Horning (1969), and many others. These works target various aspects of linguistic knowledge, but none of them address the kind of phonological knowledge that we are interested in here, and none are designed to work with OT.

We will try to show that the MDL metric, while different from current OT learners, is familiar from the evaluation criterion that a working OT phonologist might use to choose between competing hypotheses. By noting the steps that a phonologist might go through in analyzing an unknown language we will obtain a recipe for the simultaneous induction of lexicon, constraints, and ranking. We will point out that the different steps of the recipe can be unified by observing that they all involve the optimization of two quantities, one that reflects the compactness of the grammar itself (including the lexicon) and one that reflects the ease – measured through encoding length – with which the grammar can be used to describe the data. The MDL evaluation metric for the phonologist is the sum of the two quantities. After developing the MDL metric for the phonologist, we will suggest that the same criterion can form the basis for an evaluation metric for the learner.

In section 3 we will present the learning model in detail along with simulation results. We will demonstrate, using four different datasets generated by artificial grammars, that the MDL evaluation metric enables the successful learning of nontrivial combinations of lexicons and constraints. Our main result is that the evaluation metric supports the induction of lexicons and constraint rankings, aspects of the learning task that are required under all versions of OT.

The generality of the metric will also allow us to learn additional parts of the grammar without changing our learner. We will demonstrate this by learning not just the lexicon and the ranking of the constraints but also the constraints themselves. Here not all theories agree that the relevant knowledge is learned – indeed, classical OT assumes that the content of the constraints is innate. However, recent work by Heinz (2007) and Hayes and Wilson (2008) has shown that the acquisition of phonotactic knowledge is a rich and interesting question, and we believe that learning the content of the OT constraints (both markedness and faithfulness constraints) from general constraint schemata is at the very least a direction worth exploring. The learner that we present succeeds in obtaining this knowledge, combining lexical learning with the induction of specific markedness and faithfulness constraints, making it a first in this domain as well.

These learning results do require that some traditional aspects of OT analysis and learning be reconsidered. In particular, as we explain, the principles of Richness of the Base and Lexicon Optimization do not follow from the MDL metric and are left aside.

In section 4 we review previous proposals for learning within OT. As we discuss in section 4.1, most of the work in the literature focuses on questions that are quite distinct from those of the present paper. We then turn to two approaches that are much closer to our own in their aims: Maximum-Likelihood Learning of Lexicons and Grammars (Jarosz, 2006b,a), section 4.2; and Lexical-Entropy Learning (Riggle, 2006), section 4.3. We will show that these proposals can be understood in terms of the MDL evaluation metric and that this perspective highlights inherent difficulties for each of the proposals: Jarosz's approach favors grammars that describe the data well but does not take into account the compactness of the grammar itself, while Riggle's approach often favors compact grammars but does not take into account how well they describe the data. Section 5 discusses remaining questions. Section 6 concludes.

2 Evaluating Phonological Patterns using Description Length

In this section we will develop, in several, mostly informal steps, the general proposal that will serve as the basis for the concrete learner presented in formal detail in section 3. Section 2.1 provides an informal tour of how a phonologist might construct an OT analysis of an unfamiliar language. Section 2.2 shows how the choices of the phonologist can be quantified, following Solomonoff (1964) and much subsequent work on MDL. Section 2.3 provides a brief discussion of where the learner is similar to and where it is different from a phonologist comparing hypotheses. We will suggest that, despite considerable differences between the phonologist and the learner, the view of the child as a scientist searching through the grammars made available to it by UG and comparing them using the MDL criterion is a reasonable view. Readers familiar with the MDL approach to learning may wish to proceed directly to section 3.

2.1 *ab*-nese: An Informal Example

Consider a phonologist faced with the task of analyzing a newly discovered language. Suppose that the phonologist is working with an informant, who produces the following strings:

(1) *bab*, *aabab*, *ab*, *baab*, *babaaaa*, *babababababaabab*, *aaab*, *babababaa*, *babaaaa*, *aaab*, *babababababaabab*, *baab*, *bab*, *ab*, *aabab*, *aabab*, *baab*, *babababababaabab*, *aaab*, *babababaa*, *ab*, *babaaaa*, *bab*, *aaab*, *ab*, *aaab*, *aabab*, *babababaa*, *baab*

Ahead of examining the data in (1), the phonologist might take an uncommitted stance according to which any sequence of humanly pronounceable segments of a given length is equally plausible. After a quick glance at the data, however, the phonologist is struck by the following observation: of all the phonetically realizable segments, only *a*'s and *b*'s appear in the strings produced by the informant. This can be seen as an overgeneration problem for the preliminary, uncommitted hypothesis: in the absence of anything within the grammar to rule out the appearance of segments such as *c*, *d*, and *e*, their absence from (1) has to be treated as a surprising accident. The phonologist concludes that this absence is not an accident and that the new language, call it *ab*-nese, prohibits any segment other than a or b. Within the framework of OT, this restriction can be expressed by positing markedness constraints of the form $*c$, $*d$, $*e$, etc., which we will abbreviate as follows:[1]

(2) Constraints: $*\neg\{a, b\}$

The phonologist may wish to support (2) by running experiments of various sorts. For example, the phonologist may confront the speaker with two novel forms, one composed only of *a*'s and *b*'s and the other including some other segment as well. To keep the discussion simple, let us assume that if the phonologist runs such experiments then, both here and in what follows, the results support the generalizations made so far.

The constraints in (2) correctly rule out any string that includes segments other than *a* or *b*, thus solving the initial overgeneration problem. As it stands, however, the analysis in (2) still overgenerates: in the sequence in (1), certain sequences of *a*'s and *b*'s, such as *ab* and *babababaa*, appear multiple times, while other sequences of *a*'s and *b*'s, such as *baba* and *abb*, never appear at all, despite being fully compatible with (2). The strings that repeat themselves are the following:

(3)	1)	ab	3)	aaab	5)	baab	7)	babababaa
	2)	bab	4)	aabab	6)	babaaaa	8)	babababababaabab

To remedy this second overgeneration problem, the phonologist conjectures that the grammar of *ab*-nese includes a *lexicon*, a repository for information about the specific forms that are allowed. As a simple starting point, the phonologist posits (3) as the lexicon. Within the framework of OT, restricting the grammar to forms generated from a lexicon does not immediately address the overgeneration problem: selections from the lexicon can, in principle at least, surface as any form; a single entry in the lexicon can thus generate any conceivable output. In order to ensure that this does not happen and that the elements selected from the lexicon surface unchanged, the phonologist also posits a constraint, FAITH, that penalizes any changes between the chosen underlying form and its surface form:

(4) Constraints: $*\neg\{a, b\}$, FAITH

Given the lexicon in (3), the constraints in (4) are unviolated, and so no ranking among them is needed at this point. Note that when we introduced $*\neg\{a, b\}$ in (2) above, its purpose was to address the initial overgeneration problem that we encountered. Now, with the introduction of the lexicon and of an undominated FAITH, this problem is resolved independently of $*\neg\{a, b\}$. This does not mean that $*\neg\{a, b\}$ has become redundant, however: if the phonologist fails to take the restriction on the segmental inventory in *ab*-nese into account, the fact that the lexicon is written only in *a*'s and *b*'s will have to be taken to be a surprising accident; with the commitment to $*\neg\{a, b\}$, on the other hand, the lexicon seems much more natural. In other words, the present step involves a subtle but significant shift in the role of $*\neg\{a, b\}$ from an aid in making the raw data look more natural to an aid in making the lexicon look more natural.[2]

With the aid of (4), the phonologist now has a grammar that accounts for the fact that the data in (1) are instances of the entries in (3). The analysis is not fully satisfactory, however: it misses what seems like a significant generalization, namely that two *b*'s never appear in a row in a surface form. The phonologist characterizes the generalization in terms of an additional markedness constraint, **bb*, which is ranked above FAITH to ensure that *bb* sequences in the lexicon will not survive the mapping to surface forms:

(5) Constraints: $*\neg\{a, b\}$, $*bb \gg$ FAITH

Given the new markedness constraint $*bb$ and its ranking above FAITH, as in (5), an observed surface form such as *aabab* can now be generated by infinitely many underlying representations (URs). It can be generated, as before, from the faithful UR */aabab/* (6a). But it can also be generated from the UR $/aabb/$, which would violate $*bb$ if it surfaced unchanged, via a-epenthesis, as in (6b). And it can be generated from any of the infinitely many URs of the form schematized in (6c), which again would violate $*bb$, via b-deletion.

(6) Possible sources for the surface form *aabab*:

a. Faithful UR: */aabab/*

b. a-epenthesis $bb \rightarrow bab$ $/aabb/$

c. b-deletion $bb \to b$ $\{/aab^n ab^m/ : n, m \geq 1\}$

Given the new possibilities for URs that generate the observed surface forms via a-epenthesis and b-deletion using the constraint ranking in (5), the phonologist can now consider infinitely many different lexicons in addition to the fully faithful one in (3). However, while deletion and epenthesis are both possible in principle, epenthesis seems the more natural of the two and will presumably serve as the default analysis for all the cases above.

It is important to note that the preference for epenthesis over deletion of faithful URs in the cases above is no more than a default, and that it can be either bolstered or weakened by future observations of possible correlations between the segments in question and other patterns. Suppose, for example, that a closer look at *ab*-nese revealed a pattern of lengthening that generally affects the penultimate segment.[3] If that were the case, the *a*-epenthesis analysis would be supported if it turned out that *aabab* was actually $[aab{:}ab]$ – the lengthening of the antepenultimate segment could then be seen as penultimate lengthening that ignores the epenthetic *a*. The otherwise dispreferred *b*-deletion analysis would be supported if it turned out that the relevant form was [*aabab:*] – the lengthening of the final segment could then be seen as penultimate lengthening applying to the UR $/aababb/$. And an analysis that posits an underlying $/aabab/$ and does not use *bb to change the lexicon would be supported if the form was [*aaba:b*].

Informally, the default preference for epenthesis in *ab*-nese follows from considerations of economy: on the assumption that *a* is epenthesized between two adjacent *b*'s, the lexicon is smaller than it is on the assumption that it contains additional *b*'s that are deleted (or that a surface form like $aabab$ is stored as the faithful UR $/aabab/$). The underlying forms, then, are as follows:

(7)	1)	/ab/	3)	/aaab/	5)	/baab/	7)	/bbbbaa/
	2)	/bb/	4)	/aabb/	6)	/bbaaaa/	8)	/bbbbbbaabb/

The lexicon in (7) is an improvement over (3): an intuitively significant regularity, namely the absence of two consecutive *b*'s, is no longer stated as an accident of the lexicon (as it was in (3)) but is instead derived systematically by the constraints, leaving the lexicon simpler and with fewer regularities than before. Note, however, that the constraints in (5) do not allow us to take full advantage of the improved lexicon. The ranking of $*bb$ over FAITH allows us to correctly generate all of the observed forms, using *a*-epenthesis where needed, but it also allows us to employ *b*-deletion and map URs including the illicit sequence *bb* onto other, unattested forms. For example, the UR *bb* can be mapped either to the attested *bab* (through epenthesis) or to the unattested *b* (through deletion).

(8)

	/bb/	$*\neg\{a,b\}$ $*bb$	FAITH
a.	bb	*!	
b. ☛	b		*
c. ☞	bab		*

In other words, by economizing the lexicon we have introduced a new overgeneration problem.

Fortunately, the new overgeneration problem can be resolved at the cost of a very minimal further complication of the grammar. To ensure that only *a*-epenthesis resolves double-*b* sequences, we can split FAITH into two faithfulness constraints: MAX, which penalizes deletions; and DEP, which penalizes insertions.[4] We can now rank $*bb$ above DEP but not above MAX, ensuring that avoiding *bb* will justify insertion (of *a*) but not deletion (of *b*):

(9) Constraints: $*\neg\{a,b\}$, MAX, $*bb \gg$ DEP

Is the analysis complete? The answer is yes, but it will be useful to understand why. The steps we took in developing the analysis above were meant to address two kinds of concerns: we wanted to minimize overgeneration with respect to the attested forms; and we wanted to avoid any pointless complexity in the analysis itself. Let us call the first consideration *restrictiveness* and the second *economy*. As far as the data in (1) are concerned, the analysis, combining the lexicon in (7) and the constraints in (9), seems fully restrictive: it can generate only those forms that have been observed. What about economy? We just saw that _*bb_ allowed us to obtain a more compact theory. But this was just one among many patterns in the data, and it might seem tempting to try to capture some of the additional patterns as well. For example, the number of *b*'s in the examples happens to always be a power of 2: 1, 2, 4, and 8 (higher powers are missing). And the number of *a*'s is always a Fibonacci number: 1, 2, 3, 5, and 8 (all Fibonacci numbers higher than 8 are missing). Somewhat less exotically, the sequence *aaaaa* never appears in the data, and the sequence *ba* never appears word-finally. In principle at least, there is nothing to prevent us from modifying the grammar so as to take advantage of these patterns and squeeze them out of the lexicon.

For example, we could add the following markedness constraints to the grammar and rank all of them above DEP: FIB(a) (penalizing any form in which the number of *a*'s is not a Fibonacci number), 2^n(b) (penalizing any form in which the number of *b*'s is not a power of 2), **ba#*, and **aaaaa*. We can use these constraints to obtain a shorter UR for the surface form *aabab*.

(10)

	/aaba/	$*\neg\{a,b\}$	MAX	$*bb$	FIB(a)	2^n(b)	**ba#*	**aaaaa*	DEP
a.	aaba						*!		
b.	aab		*!						
c.	aabaa				*!				*
d. ☞	aabab								*

In order to save a single segment in this UR, we needed two new constraints – FIB(a) and **ba#* (the remaining two new constraints were not involved in this case). This tradeoff is hardly a bargain, and it does not improve much through consideration of the remaining forms in the lexicon. If enough additional forms of the general pattern exhibited by *aabab* are encountered, the resulting savings in the storage of the URs will justify the price paid by introducing the new constraints. But for now, these facts, and infinitely many additional ones, do not help make the analysis simpler and are best treated as accidents rather than meaningful patterns: the analysis, as far as the current data – and the goal of minimizing both economy and restrictiveness – are concerned, is complete.

2.2 An Evaluation Metric for the Phonologist

2.2.1 Analyzing *ab*-nese using Description Length

The process just described attempts to maximize the economy and the restrictiveness of the grammar given the data. In section 2.3 we will use the phonologist's criterion for comparing hypotheses – the phonologist's evaluation metric – as a model for the learner's evaluation metric. Before we can do that, however, we will need to make the phonologist's evaluation metric more explicit. In particular, we will need to understand how economy and restrictiveness are measured and how the two measurements are combined. As it turns out, it is easy to make incorrect choices here, choices that would lead the phonologist to favor hypotheses that clash directly with our intuitions regarding linguistic analysis. We will see a few illustrative cases below. But let us start with what we think is the right choice, first formulated by Ray Solomonoff (Solomonoff, 1960, 1964). According to Solomonoff, a hypothesis is a complete description of the data – think of it as a computer program that runs, prints out the data, and then halts. The goodness of a hypothesis is determined by its length: the shorter the hypothesis (for example, as measured in bits in the source file containing it on the computer), the better it is. It is often convenient to separate the logic of the program from any accidental aspects of the data and think of the program as the combination of two distinct parts: the 'real' program, or grammar, which we will write as G; and the encoding of the data D using the grammar, which we will write as D:G. As we will shortly see, the length of G, $|G|$, corresponds to the informal notion of economy, while the length of D:G, $|D{:}G|$, corresponds to restrictiveness. The goal of the phonologist, on this view, is to find the hypothesis that provides the shortest overall length. That is, the grammar that provides the shortest value for the sum $|G| + |D{:}G|$.

$$\underbrace{\underbrace{010101101010101001010}_{Lexicon}\underbrace{10101010010}_{Constraints}}_{G}\underbrace{10100010110101}_{D:G}$$

Figure 1: Schematic view of Solomonoff's evaluation metric as applied to OT. The grammar G consists of both lexicon and constraints. (The bit string in this figure is notional and is only intended as a schematic illustration of how some G can be represented using the guidelines discussed in the present section; concrete examples are discussed in detail in section 3.) The data D are represented not directly but as encoded by G. The overall description of the data is the combination of G and D:G.

Let us illustrate. Suppose we wish to obtain a complete description of the data in (1), for example in order to convey it to a phonologist who has no direct access to our informant. Ahead of the analysis that we went through, the data would be no more than an arbitrary sequence to us. To convey it we can do no better than transmit it symbol by symbol, specifying at each step which symbol is chosen out of the full alphabet. The usual way of specifying choices out of a set is as a string of bits – that is, a string of binary choices, each of which can be 0 or 1. If the full alphabet has four elements, for example, we can arrange them in a row – say, a_1, a_2, a_3, and a_4 – and specify the choice using two bits: the first specifying whether the choice is among the leftmost two or the rightmost two (so 0 says that the choice is among a_1 and a_2 and 1 says that the choice is among a_3 and a_4) and the second doing the same within the subset specified by the first (so if the first bit was 0 and the second bit was 1, then the specified element is a_2). If there are eight elements in our full alphabet, write them as a_1 to a_8, two bits would no longer suffice: we would need an additional bit to specify first whether the chosen element is among the leftmost four or the rightmost four, after which two bits will allow us to specify the exact choice as before. More generally, if there are n elements in our full alphabet, we would need $\lceil \lg n \rceil$ bits to specify an individual element. For example, if our alphabet is the IPA, which has 107 letters and 31 diacritics, we would need $\lceil \lg(107 + 31) \rceil = 8$ bits to encode an individual choice. To convey the data in (1) under the null hypothesis, then, we would need to spend the number of bits we require to encode an arbitrary symbol – eight if we are using the IPA – times the number of characters in the sequence, including commas.

As soon as we notice that only *a*'s, *b*'s, and commas occur in the input data, we can replace the eight bits per symbol with a fixed code length of two bits per symbol, and the length of the code drops accordingly.[5] Encoding the restriction of the segmental inventory to the set $\{a, b\}$ takes up a few additional bits, thus increasing $|G|$ slightly, but this addition is easily offset by the savings to $|D{:}G|$ obtained through the drop from eight bits per symbol to two, even for a relatively short text.[6] Note that the phonologist's notion of overgeneration – that is, of a hypothesis being overly inclusive, making the attested data seem atypical (and thus surprising) – translates into a statement about $D{:}G$ being too long. The comparison between the two hypotheses is schematized in Figure 2.

Our next step in the analysis, introducing a lexicon, allows us to derive further savings. If there are only eight sequences that keep repeating themselves, we no longer need to encode each segment

$$\underbrace{10101010010}_{Null\ G}\underbrace{\underbrace{10100010}_{8}\underbrace{11001011}_{8}\underbrace{11001011}_{8}\underbrace{10100010}_{8}\underbrace{10100010}_{8}\ldots}_{D:G}$$

$$\underbrace{01101010010111001110}_{G=*\neg a,b}\underbrace{\underbrace{10}_{2}\underbrace{11}_{2}\underbrace{11}_{2}\underbrace{10}_{2}\underbrace{10}_{2}\ldots}_{D:G}$$

Figure 2: Two simple hypotheses (schematic). The null hypothesis (top) treats the data as an arbitrary sequence of segments. Encoding the grammar is simple, but the price paid for encoding the data is high: eight bits per segment. The hypothesis that treats the data as an arbitrary sequence of a's, b's, and commas requires a slightly more complex grammar, but the savings in encoding the data are noticeable: we now have to pay only two bits per segment.

individually. Instead, we can transmit the lexicon once, in the beginning of the transmission, and then use $\lg 8 = 3$ bits to specify which word is chosen each time. For *babaaaa*, for example, this would mean three bits instead of fourteen bits for each occurrence of the word.

Observing that sequences of the form *bb* are systematically absent allows us to compress the lexicon introduced in the previous step: we increase the size of the grammar slightly, by adding the constraint **bb*, and this allows us to decrease the overall size of the grammar by removing inter-*b* instances of *a*. Note that this trade-off is carried out entirely at the level of economy; that is, in terms of $|G|$ (we will immediately turn to the effect of this move on restrictiveness; that is, $|D{:}G|$). Here the savings are not as dramatic as they were in the previous steps, though they might still be meaningful, and they would be even more so with a bigger lexicon (assuming it conformed to the same pattern).

Next, as long as FAITH is kept as an atomic constraint, resolving an underlying *bb* sequence through *a*-insertion and through *b*-deletion would incur the same violation marks. This, in turn, leads to an overgeneration problem that would leave us worse off than with the uncompressed lexicon: each time the UR *bb* is selected in order to produce the surface form *bab*, the system so far would generate two winning candidates, the attested *bab* and the unattested *b*. Again, the phonologist's notion of overgeneration translates into an overly long $D{:}G$. We would thus have to spend additional bits to ensure that we produce the former and not the latter. We overcome this problem by splitting FAITH into two separate constraints, MAX and DEP, and by ranking **bb* above the latter but not above the former. The splitting of FAITH slightly increases the size of the grammar, but it is a one-time increase, and after that every time the UR *bb* is selected, it will lead deterministically to the surface form *bab*. The past three steps are schematized in Figure 3.

$$\underbrace{\underbrace{11010011010011101101}_{Lex=\text{naive}}\ \underbrace{101110}_{Con=\textsc{Faith}}}_{G}\ \underbrace{\underbrace{010}_{3}\underbrace{110}_{3}\underbrace{011}_{3}\underbrace{011}_{3}\underbrace{010}_{3}\underbrace{110}_{3}\underbrace{011}_{3}\ldots}_{D:G}$$

$$\underbrace{\underbrace{010110010001}_{Lex=compressed}\ \underbrace{1011101001}_{Con=\textsc{Faith},*bb}}_{G}\ \underbrace{\underbrace{010\ 0}_{4}\underbrace{110}_{3}\underbrace{011\ 0}_{4}\underbrace{011\ 0}_{4}\underbrace{010\ 0}_{4}\underbrace{110}_{3}\underbrace{011\ 0}_{4}\ldots}_{D:G}$$

$$\underbrace{\underbrace{010110010001}_{Lex=compressed}\ \underbrace{10011011110110}_{Con=\textsc{Max},*bb\gg\textsc{Dep}}}_{G}\ \underbrace{\underbrace{010}_{3}\underbrace{110}_{3}\underbrace{011}_{3}\underbrace{011}_{3}\underbrace{010}_{3}\underbrace{110}_{3}\underbrace{011}_{3}\ldots}_{D:G}$$

Figure 3: Three more advanced hypotheses. Introducing a naive lexicon, in which the attested strings are listed, allows us to describe the data word-by-word rather than segment-by-segment, yielding significant savings (top). Squeezing the pattern **bb* out of the lexicon results in a shorter lexicon but longer overall description length: for each UR that includes the sequence *bb*, we will now need to specify that the surface form is the result of *a*-epenthesis rather than *b*-deletion (middle). Splitting FAITH into MAX and DEP allows us to maintain both a short lexicon and a short description of the data at the modest cost of a slight complication of the constraints, leading to the shortest overall length (bottom).

Finally, the putative patterns of powers of 2 and the Fibonacci sequence seem quite unhelpful at this point in terms of compression, as do $*a^5$ and $*ba\#$. Differently from $*bb$, which aided in compression and was thus taken to capture a meaningful gap, these other patterns would be taken by the phonologist to capture accidental gaps – from the perspective of description length, capturing these patterns lengthens $|G|$ more than it shortens $|D{:}G|$ – and consequently they are not added to the grammar.

2.2.2 Economy and Restrictiveness must be Minimized Together

Each of our steps above attempted to improve the analysis by shortening the encoding. In this respect, the phonologist's strategy is one among many imaginable strategies incorporating a simplicity bias, a general approach that is often associated with Occam's Razor. But the details of how simplicity is implemented matter. Crucially, what matters to the phonologist is the *entire* message length: economy (that is, the length of the grammar, including the lexicon) and restrictiveness (that is, the encoding of the data given the grammar) must be balanced against one another. Minimizing only the one or only the other would lead to unsatisfactory results, as we now discuss.

Suppose, for example, that the phonologist had ignored restrictiveness and focused on economy alone. The phonologist would then never have departed from the initial, perfectly simple hypothesis that said that any sequence of segments is possible. And if forced by someone to abandon that hypothesis and accept that only *a*'s and *b*'s occurred, the phonologist would have settled on that hypothesis and moved no further. If forced to move forward and adopt a lexicon, the phonologist might have had an incentive to minimize it by adding $*bb$ to the grammar and shortening the URs, but they would have had no cause to split FAITH into MAX and DEP. In each of these steps, we have a simple but incorrect hypothesis that admits a proper superset of *ab*-nese. The *ab*-nese data will of course never furnish a counterexample to such a hypothesis – an instance of the so-called *subset problem* – and the exclusive focus on economy will leave the phonologist with the incorrect superset language. Since the simpler hypothesis in these cases is overly inclusive, it will need to be able to encode not only the elements of *ab*-nese but also those elements of the superset language that are not in *ab*-nese, such as *b*, which can result in an encoding of the *ab*-nese data that is considerably longer than under a more restrictive hypothesis that does not need to be able to encode elements such as *b*. In other words, an exclusive focus on economy can lead to a lengthening of $|D{:}G|$ that more than offsets any gains in $|G|$. Note that combining a first step of economy with a second step of restrictiveness will be of little help: the problematic winner in each of the steps just summarized is strictly simpler than the losing competitor, thus making it impossible for some tie-breaking criterion in the second step to reverse the overgeneration problem. Economy alone is the essence of the evaluation metric of Chomsky and Halle (1968)'s SPE, and a two-step architecture in which a criterion such as restrictiveness operates on the outcome of economy is at the heart of the earlier version of the evaluation metric in Chomsky (1951), as well as what Kiparsky (2007) calls Pāṇini's Razor.[7] The problem for economy has been noted by Braine (1971), Baker (1979), and Dell (1981), and we will revisit it in our discussion of Riggle (2006)'s Lexical Entropy learner in section 4.3.

Consider next what would happen if the phonologist chose to ignore economy and focus on restrictiveness alone. In particular, it is sometimes suggested that, as a remedy to the subset problem, generalization should be conservative and always choose the smallest language under consideration that is compatible with the data, a preference known as the *subset principle* (see Wexler and Culicover, 1980, Berwick, 1985, and Manzini and Wexler, 1987; for recent discussions of the subset problem in phonology see Albright and Hayes 2011 and Heinz and Riggle 2011). From the perspective of description length, restrictiveness alone can be implemented as a preference for shortening $D{:}G$ (that is, a preference for a grammar that makes the data typical), irrespective of $|G|$. Restrictiveness alone is an approach that respects the subset principle. While escaping the subset problem, a phonologist relying on the subset principle runs straight into the mirror image of the problem for economy alone: instead of wild overgeneralization, such a phonologist never generalizes at all. In the case of *ab*-nese, for example, a phonologist focusing on restrictiveness alone would have been perfectly content with our first lexicon, which simply memorizes the surface forms, with no incentive to add $*bb$ and compress the URs. This, in turn, would make a putative future word *aab* equally easy to accommodate as *abb*; and while *ab*-nese is of course artificial, the counterparts of this prediction for natural languages such as English have been recognized as problematic as early as Halle (1962). We will revisit this prediction in our discussion of Jarosz (2006b,a)'s Maximum Likelihood learner in section 4.2 below. The dangers of adhering to the subset principle become particularly clear when the language is infinite (or just too big for the phonologist to encounter in its entirety). To keep things simple, imagine a dialect of *ab*-nese, call it *zab*-nese, in which any nonnegative number of *z*'s can precede any word. We would expect a reasonable phonologist to notice this generalization after enough surface forms have been observed. A fully restrictive phonologist, however, will never generalize. At any given point, such a phonologist will have had exposure to a finite number of such *z*-variants, and these forms will be listed as part of the grammar, thus increasing $|G|$ with each newly observed form. So while the gains of economy alone in $|G|$ often lead to the lengthening of $D{:}G$, the gains of restrictiveness alone in $|D{:}G|$ often arise through memorization of the data in G, which can result in considerable lengthening of $|G|$. Note also that, as with our earlier discussion of economy, the problem will not be solved by using restrictiveness as a first step that then feeds a second criterion such as simplicity. The incorrect winner at each step in the case of *zab*-nese will always be strictly more restrictive than the correct hypothesis, rendering a tie-breaking second step useless.

In short, we must take both economy and restrictiveness into account, and we must minimize both simultaneously: a good hypothesis is one that balances the minimization of $|G|$ (which favors simple but often overly inclusive hypotheses) with that of $|D{:}G|$ (which favors restrictive but often overly memorized hypotheses). Using the MDL value $|G| + |D{:}G|$ as the evaluation metric provides exactly the right kind of balance. As mentioned, the first to propose this idea was Solomonoff (1960, 1964), who used his discovery to formulate a fully general theory of prediction. The same idea of viewing hypotheses as programs that output the data and defining their value according to their length was discovered independently (from a slightly different perspective) by Kolmogorov (1965) and Chaitin (1966). The length of the shortest program that outputs the data D is known as the *Kolmogorov complexity* of D and is written $K(D)$.[8] Kolmogorov complexity is not computable, and while it is an important tool for deriving results about learnability in principle, as in Chater and Vitányi (2007), it is often necessary to restrict the hypothesis space to ensure computability. This is done in the frameworks of Minimum Message Length (MML; Wallace and Boulton, 1968) and Minimum Description Length (MDL; Rissanen, 1978). To simplify terminology, and since the differences between the frameworks incorporating Solomonoff's insight will not be central to our proposal, we will refer to any attempt to minimize $|G| + |D{:}G|$ (often within a restricted family of possible grammars) as MDL. The relevance of MDL for grammar induction was already noted by Solomonoff (1964). Over the years, several authors have used MDL profitably for grammar induction, either as a methodological principle for the scientist or as a learning criterion for the learner. Notable examples include Berwick (1982), Rissanen and Ristad (1994), Stolcke (1994), Brent and Cartwright (1996), Grünwald (1996), de Marcken (1996), Clark (2001), Goldsmith (2001, 2010), Dowman (2007), Hsu and Chater (2010), Hsu et al. (2011), and

Goldsmith and Riggle (2012).

In section 1 we mentioned that MDL has not featured centrally in works on acquiring phonological knowledge. In the literature on learning in OT, the guiding principles (especially Richness of the Base and Lexicon Optimization) are quite different from MDL. Having tried to show how MDL arises naturally from the perspective of the working phonologist, let us explain why the same criterion can make sense for the child learner. With that background we will then proceed to present our MDL learner in section 3.

2.3 From Phonologist to Learner

A learner is not a phonologist. The phonologist may, in principle at least, consider any program as a grammar; the learner, on the other hand, may well be restricted by UG to a very limited search space. Earlier, for example, we suggested that the phonologist may consider – but ultimately reject – four patterns in the *ab*-nese data: the number of *a*'s is always a Fibonacci number; the number of *b*'s is always a power of two; the sequence *aaaaa* never occurs; and the sequence *ba* never occurs word-finally. While it is conceivable that the child learner is also capable of entertaining all these patterns, it could also be that some of these patterns are impossible for the child to represent, or, as discussed by Heinz (2007), the child might be able to represent certain patterns but incapable of reaching them through its learning procedure. The differences between the learner and an ideal scientist are the focus of a growing literature on underlearning, which investigates the limitations of what humans can learn. See Smith (1966), Peña et al. (2002), Endress et al. (2007), Moreton (2008), Endress et al. (2009), Endress and Mehler (2010), and Becker et al. (2011), among others.

There are other differences as well. For example, the phonologist and the child differ in the degree of control each has on their respective inputs: as mentioned earlier, the phonologist may run controlled experiments using a variety of methodologies, recruit typological data, and obtain systematic negative evidence; the learner, on the other hand, is largely restricted to the kinds of evidence that are given to it by its environment. And the phonologist may also record many years of data and make reference to all the information accumulated in this fashion, while the child is quite unlikely to record explicitly the entire history of speech to which it has been exposed.[9]

But in one important respect, the learner and the phonologist have a great deal in common: both face the task of making sense of unanalyzed data in the language they are immersed in, and both bring to the task a hypothesis space, each point in which represents a grammar. Not all the grammars in the hypothesis space will be able to generate the data in the first place, but for any grammar that can, we can look for a sequence of instructions to the grammar – a key – that will generate exactly the part of the data that we have seen. As discussed above, a message consisting of the combination of a grammar and a key provides a full description of the data, and we can think of the phonologist as searching the space for the grammar that yields the shortest such message. The hypothesis space for the phonologist is biased toward mechanisms that work well with past observations – recall that in discussing *ab*-nese we took it for granted that we could easily encode constraints, ranking, and lexicons – but it is a very big space, and it includes many additional mechanisms (for example, if *ab*-nese turned out to be problematic for OT, we might consider a complete revision of the architectural premises of the grammar). This was the essence of the discovery procedure that we built in the previous section.

For the child learner, things are less clear. Like the phonologist, the child attempts to settle on a point in the hypothesis space (in the case of the child, the hypothesis space is probably considerably more limited than for the phonologist; for example, either OT or SPE but presumably not both). But there is little conclusive evidence to date about how the child chooses this point. It is conceivable, of course, that the child searches through its hypothesis space for the hypothesis that yields the most favorable value for some evaluation metric, and it is conceivable that this metric is MDL; but there are any number of other methods that the child might be using, and many of them do not amount to optimization using an evaluation metric at all. For example, it might look for the first grammar under some enumeration that is compatible with the data; or it might look for a grammar that is *not* compatible with the data; or it might use the first two words in the input data as a key for selecting a grammar out of a big table, regardless of compatibility; and so on. Of course, there are also learning procedures that are considerably more reasonable than those just mentioned. See, in particular, Manzini and Wexler (1987), Gibson and Wexler (1994), and Niyogi and Berwick (1996) for such procedures for the hypothesis space provided by the Principles and Parameters framework of Chomsky (1981); in section 4 we will review several procedures of this kind that have been proposed for the hypothesis space provided by OT. See Chomsky and Halle (1968), Braine (1971), Wexler and Culicover (1980), and Jain et al. (1999) for further discussion of the child and its space of hypotheses.

In section 3 below we will present simulations that show that show successful learning of several phonological patterns by the child-as-phonologist model, suggesting that this model is at the very least a viable approach to phonological learning. Before doing so, we wish to highlight a perspective from which the child-as-phonologist model, when applied to the hypothesis space provided by UG and with the MDL evaluation metric, is a particularly natural approach (see Katzir 2014 for an elaboration of this perspective). The child already has access to the hypothesis space, and each point in that space that allows it to parse the data provides the basis for at least one full description of the data; all that is missing is the ability to traverse this space and test different hypotheses, comparing the messages they support in terms of overall length. If the child can maintain a current hypothesis and a new hypothesis simultaneously and use them both to parse the data, and if the child can switch from one hypothesis to another in a way that lets it traverse a portion of the hypothesis space that allows convergence, it will be able to mirror the phonologist's search. And if the child can compare the overall memory space required to encode the data using two hypotheses, it can mirror the phonologist's criterion. The procedure that parallels the phonologist, then, is available to the child almost in full simply by virtue of having the ability to represent and use grammars from within the set allowed by UG: indeed, it seems that one would have to make special stipulations to block such a procedure. If this perspective is right, then, it makes sense to take the child-as-phonologist model as a methodological starting point.

3 Simulation Results

We now proceed to present (in the current section) evidence that a learner based on the model of the child as MDL phonologist can succeed on linguistically relevant patterns. Then, in section 4, we will discuss alternative learners proposed in the literature on learning in OT and attempt to show that these learners are less successful than our MDL learner in handling patterns of the kind discussed here.

We will not be able to test the learner on a real-life corpus at this point. Instead, we will provide a proof-of-concept demonstration, using datasets generated by artificial grammars that incorporate phonologically interesting dependencies.[10] Section 3.1 presents the general setting for our learning simulations, including the details of how grammars are encoded, how they are used to encode the data, and how the search is performed. We then present simulation results for the *ab*-nese dataset (section 3.2), move on to a language that exhibits some phonological patterns familiar from aspiration in English (section 3.3), continue to a dataset showing restricted optionality along the lines of *l*-deletion in French (section 3.4), and end with a dataset exemplifying the ability of the current approach to learn from alternations, modeled after voicing assimilation in Modern Hebrew (section 3.5). As we will see, the learner extracts grammars that seem phonologically appropriate in all four cases. Before examining the simulations, let us present the setting.

3.1 The Setting

3.1.1 Encoding

We need to commit in advance to the search space defined by UG. We will assume that this space is defined by: (a) the ability to state lexicons using a fixed alphabet of feature vectors; and (b) the ability to state constraints and their ranking – which we take to always be a total ordering – using two kinds of very general constraint schemata, one for faithfulness constraints and one for markedness constraints, as shown in Figure 4.[11] We wish to emphasize, though, that our goal is not to argue for this particular theory of UG over other theories; rather, it is to demonstrate how learning can take place given a search space provided by UG and our evaluation metric.

$$\textsc{Dep}(F) \quad \textsc{Max}(F) \quad \textsc{Ident}(F) \quad {}^{*}F_1F_2\ldots F_n$$

Figure 4: Constraint schemata available to the learner. F's represent feature bundles.

Recall that our goal is to encode hypotheses as fully explicit messages – specifically, as binary strings – and compare them according to their lengths. Once an encoding scheme is chosen, each grammar G in the search space is associated with a value $|G|+|D{:}G|$ that is obtained by combining the description length of G itself (the lexicon and the constraints) and the description length of the data given G. In the present subsection, we will consider one simple – and by no means optimal – encoding scheme, based on the feature table in (11), for the binary features *consonantal* and *continuant*; we assume that the table is given to the learner in advance.[12] Every simulation that we present in the following sections will be accompanied by its corresponding feature table.

(11)

	a	b	s
$cons$	-	+	+
$cont$	+	-	+

Let us start with measuring the description length of the lexicon. Consider the lexicon in (12a). Using a delimiter (#) to mark the end of each word and one additional delimiter to mark the end of the lexicon, we obtain the string representation in (12b). The lexicon is encoded as a binary string by substituting a two-digit binary code for each symbol in (12b): 00 for $+$, 01 for $-$, and 10 for $\#$; given the feature table in (11), this will result in four bits per segment. The size of the lexicon will be the length of the string in (12c).

(12) a. $\{asa, ba, bsab\}$

b. $-+++-+\#+--+\#+-++-++-\#\#$

c. 01000000010010000101001000010000010000011010

We use a similar procedure to encode the constraints and their ranking (which we take to be a total ordering). The constraint hierarchy in (13a) is represented as the string (13b): the symbols

D, M, I, and P stand for DEP, MAX, IDENT, and phonotactic constraints respectively; a delimiter marks the end of each constraint; for phonotactic constraints, a delimiter marks the end of each feature bundle (the last feature bundle of a phonotactic constraint is therefore followed by two delimiters); one additional delimiter marks the end of the constraint hierarchy. We enumerate all symbols that can play a role in constraint descriptions and assign each symbol a fixed binary code as demonstrated in Figure 5. The description length associated with the constraint set is the length of the binary translation of 13b according to Figure 5.

(13) a. $\text{D}\textsc{ep}(-cons) \gg \text{M}\textsc{ax}(+cont) \gg {}^*[+cons]\begin{bmatrix}-cons\\ \\ +cont\end{bmatrix} \gg \text{I}\textsc{dent}(-cont)$

b. $D - cons\#M + cont\#P + cons\# - cons + cont\#\#I - cont\#\#$

Symbol	Code
D	0000
M	0001
I	0010
P	0011

Symbol	Code
cons	0100
cont	0101

Symbol	Code
$+$	0110
$-$	0111

Symbol	Code
#	1000

Figure 5: Binary code assigned to each symbol.

We proceed to measure the length of the data given the grammar, $|D{:}G|$. For convenience, let $s_1, \ldots, s_n$ be an enumeration of the surface representations presented as data to the learner and $u_1, \ldots, u_m$ an enumeration of the URs in G's lexicon. For every choice u_i from the lexicon, the phonological mapping defined by G returns as output a set of surface representations, the set of optimal output candidates for u_i (say $o_{i,1}, o_{i,2}, \ldots$).[13] Describing a surface representation that can be parsed by the grammar amounts to specifying two successive choices: a choice of a UR u_i from the lexicon and a choice of an optimal output $o_{i,j}$ of that UR. We assign each choice from the lexicon a fixed binary code as illustrated in (14a) (for the case $m = 5$). Choices from sets of optimal output candidates receive similar treatment (14b): given a UR, all optimal candidates are enumerated; the number of bits required to specify a choice of an optimal candidate depends on the total number of optimal candidates for the UR (in the middle table no code is needed as the choice is deterministic).

(14) a.

UR	Code
u_1	000
u_2	001
u_3	010
u_4	011
u_5	100

b.

u_1	
Output	Code
$o_{1,1}$	00
$o_{1,2}$	01
$o_{1,3}$	10

u_2	
Output	Code
$o_{2,1}$	

u_3	
Output	Code
$o_{3,1}$	0
$o_{3,2}$	1

. . .

Suppose now that we wish to encode s_1 given G. If s_1 cannot be parsed by G, there is no finite binary string that can serve as a description of s_1, and its description length would be taken to be infinite. Alternatively, suppose that s_1 is equal to the output $o_{1,3}$ in our example (14b). In that case, s_1 can be described by the binary string 00010 (000 specifies the choice of u_1, 10 the choice of $o_{1,3}$), so its description length would be 5. In general, phonological grammars are ambiguous, and it is possible that a given surface representation has more than one parse. For example, s_1 could also be equal to $o_{3,1}$, an output of u_3 under G. When there are multiple descriptions available, the shortest one will be chosen. In our example, the string 0100 would end up as the shortest description of s_1, a description of length 4.[14] We arrive at the total description of D:G by concatenating the descriptions of $s_1, \ldots, s_n$. $|D{:}G|$ is the length of the resulting concatenation. We also chose to multiply the summand $|D{:}G|$ by 100 in the simulations for *ab*-nese (section 3.2) and for aspiration (section 3.3) due to performance considerations.[15]

3.1.2 EVAL

The algorithmic infrastructure of our system is closely based on the finite-state implementation of OT developed in Riggle (2004). The constraint hierarchy is represented as an ordered list of individual constraints, each of which is implemented as a weighted finite-state transducer. The transducers are intersected to form the EVAL component of OT. The reader is referred to Riggle (2004) for details of implementation and optimization and to Heinz et al. (2009) for a discussion of the problem and its implications from the perspective of computational complexity.

The properties of EVAL are not taken into consideration in evaluating or comparing the complexity of hypotheses. In particular, the complexity of constraints as measured by our metric is blind to the size of their corresponding finite-state machines, and the correlation between the two does not seem to be very strong.

3.1.3 Search

Our focus in this paper is the learning criterion. We make no cognitive claims regarding either the search procedure or the initial state of the search. To make the learner concrete, though, we must make commitments with respect to both. For the search procedure, we adopt Simulated Annealing (SA; Kirkpatrick et al., 1983), a general strategy, schematized in Figure 6 and discussed below, which supports searching through complicated spaces that involve multiple local optima.

```
$D \leftarrow$ input string in $\Sigma$
$G \leftarrow$ initial_grammar($\Sigma$)
$T \leftarrow$ initial temperature
while $T > threshold$ do
  $G' \leftarrow random\_neighbor(G)$
  $\Delta \leftarrow [|G'| + |D{:}G'|] - [|G| + |D{:}G|]$
  if $\Delta < 0$ then
    $p \leftarrow 1$
  else
    $p \leftarrow e^{-\frac{\Delta}{T}}$
  end if
  choose $G \leftarrow G'$ with probability $p$
  $T \leftarrow \alpha T$
end while
return $G$
```

Figure 6: Pseudocode of the search procedure.

SA proceeds by comparing a current hypothesis to its neighbors in terms of their goodness, which in our case is the total description length. That is if a current hypothesis G has G' as its neighbor, $|G| + |D{:}G|$ is compared to $|G'| + |D{:}G'|$. If G' is better than G, the search switches to G'. Otherwise, the choice of whether to switch to G' is made probabilistically and depends both on how much worse G' is and on a *temperature* parameter. The higher the temperature, the more likely the search is to switch to a bad neighbor. The temperature is initially set to a relatively high value, and it is gradually lowered as the search progresses, making the search increasingly greedy. Our initial temperature was 100, and it was lowered according to a cooling schedule in which the temperature at each step is multiplied by a constant $\alpha = 0.999985$ to yield the temperature at the next step. The search ends when the temperature descends below a threshold of 0.01.[16] We have not yet conducted a systematic study to determine how robust the results reported below are with respect to different choices of the search parameters. Again, we stress that our interest is the evaluation metric and not the search, regarding which we make no cognitive claims.

For the initial state, we assume the naive one in which no patterns in the data have been discovered. The grammar includes a single faithfulness constraint FAITH that penalizes any structural change, thus enforcing an identity mapping between URs and surface forms; the lexicon in the initial grammar is a list of the surface forms in the input data.[17] FAITH is included as an additional symbol in the calculation of the size of the constraint set (Figure 5 above). For any grammar G, the neighbor grammar G' is generated as a variant of G in which one of the changes in (15) occurs.

(15) a. A segment is added in the lexicon.

b. A segment is removed from the lexicon.

c. A segment is modified in the lexicon.

d. A constraint with a single feature bundle is added in the constraint hierarchy.

e. A constraint is removed from the constraint hierarchy.

f. A constraint is demoted by one place in the constraint hierarchy.

g. A single feature bundle is added to a phonotactic constraint in the constraint hierarchy.

h. A single feature bundle is removed from a phonotactic constraint in the constraint hierarchy.

The modification is chosen according to a uniform distribution over possible changes. All decisions in a given modification are made randomly as well (positions for insertion, deletion, and demotion; feature bundles, segments, and constraints for insertion and modification). There is no upper bound on the size of the lexicon, the size of a phonotactic constraint, or the size of the constraint hierarchy.

3.2 *ab*-nese

Our first dataset is a language similar to *ab*-nese, presented in section 2.1 above and repeated here:

(16) *bab*, *aabab*, *ab*, *baab*, *babaaaa*, *babababababaabab*, *aaab*, *babababaa*, *babaaaa*, *aaab*, *babababababaabab*, *baab*, *bab*, *ab*, *aabab*, *aabab*, *baab*, *babababababaabab*, *aaab*, *babababaa*, *ab*, *babaaaa*, *bab*, *aaab*, *ab*, *aaab*, *aabab*, *babababaa*, *baab*

Given an alphabet $\Sigma = \{a, b\}$ and one feature $\pm cons$ ($a = [-cons]$, $b = [+cons]$), we generated an initial pool of words by taking all combinations of $1 - 6$ syllables from the set $\{a, ab, ba, bab\}$. We then filtered out all words that included the sequence bb and provided the learner with the resulting set of words ($n = 252$). The full input for this simulation (and the following ones) is provided in Appendix 6.As discussed in section 3.1.3 above, the initial state includes a constraint set with a single FAITH constraint and a lexicon identical to the data:

(17) Initial grammar:

$$G_{initial} = \begin{cases} \text{LEX:} & bab, aabab, ab, baab, babaaa, babababaa \ldots \\ \text{CON:} & \text{FAITH} \end{cases}$$

$$\text{Description length: } |G_{initial}| + |D{:}G_{initial}| = 4,622 + 201,600 = 206,222$$

As discussed in section 2.1, the absence of bb sequences from the data can be used to obtain a more concise description of it. Consequently, the evaluation metric favors grammars that encode this pattern over grammars that treat it as a mere accident. Our learner converged on a final hypothesis in which all relevant instances of a have been removed from the lexicon and inserted by the grammar:

(18) Final grammar:

$$G_{final} = \begin{cases} \text{LEX:} & bb, aabb, ab, baab, bbaaa, bbbbaa, \ldots \\ \text{CON:} & \text{MAX}([+cons]) \gg *[+cons][+cons] \gg \text{FAITH} \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 4,028 + 201,600 = 205,628$

The addition of both the markedness ($*[+cons][+cons]$) and the faithfulness (MAX($[+cons]$)) constraints increases the length of CON but helps in minimizing the total description length. The markedness constraint allows the learner to compress the lexicon by preventing bb sequences from surfacing. The faithfulness constraint is introduced to ensure that b deletion incurs more violations than a epenthesis. The latter option is therefore deterministically chosen for satisfying the markedness constraint, and the length of the data given the grammar becomes lower than it would have been had the faithfulness constraint been left out. The learner has converged on a simple, restrictive grammar that accords well with our intuitions about what a correct grammar for the data should look like.

Note that the result differs from the final grammar in our discussion in section 2.1 in two respects. First, a MAX constraint is added instead of having FAITH split into MAX and DEP. This occurs since DEP does not require a shorter description length than FAITH and there is no reason for the evaluation metric to favor it. The second difference is that our representations only allow strict ranking of constraints in CON and so MAX($[+cons]$) can be ranked anywhere in the hierarchy, whereas in our previous discussion we assumed that non-obligatory rankings were possible.

3.3 Aspiration

Our next dataset shows a pattern modeled after aspiration in English and is designed to test the learner on the problem of allophonic distribution. Simplifying, we assume that the ambient language has aspirated stops (like t^h and k^h) appearing before vowels but not elsewhere. The distribution of aspiration is thus entirely predictable. We expect the learner to treat aspirated stops as allophones of their unaspirated counterparts. Aspiration in the appropriate places should not be the result of accidents of the lexicon; rather, it should be enforced by the grammar. One way to enforce the correct distribution of aspiration, in line with earlier work in generative phonology but generally not with work following the OT learning principle of Lexicon Optimization, is to remove aspiration from the lexicon altogether and ensure its insertion through the constraints: the UR of $[k^h\text{æ}t]$ becomes $/k\text{æ}t/$ and the UR of $[t^hik^hit]$ becomes $/tikit/$, while surface forms where aspiration is missing in the right context (like $^*kat$) should be ungrammatical. Importantly, the grammar should also block aspiration from occurring elsewhere, as in the illicit surface forms $^*at^h$ and $^*k^hik^ht$.

Previously, we explained why the MDL evaluation metric favors grammars that treat patterns such as $*bb$ or the present ban on unaspirated prevocalic stops systematically rather than leaving them as accidents of the lexicon. Adding the relevant constraints to CON increases its description length but makes it possible to squeeze information out of the lexicon, thereby lowering the total description length. Here, blocking of aspiration in elsewhere contexts presents a further learnability challenge. The crucial point is that a grammar that generates aspirated stops before vowels is not necessarily restrictive enough; the grammar should also prevent cases where URs like $/at^h$/ or $/kik^ht/$ surface with stray aspirated segments.

One way for the learner to approach this problem is to allow forms like at^h and kik^ht to be represented underlyingly and block $^*at^h$ and $^*k^hik^ht$ as part of the input-output mapping. This direction, in line with the OT principle of Richness of the Base (ROTB), is not available to our MDL learner: on natural assumptions about the representation of aspiration, a hypothesis with additional underlying instances of aspiration will be more complex than one without them and will thus be dispreferred;[18] and in the absence of such additional instances of underlying aspiration, a constraint that ensures that they do not surface will serve no compressional purpose and will likewise cause the hypothesis to be dispreferred. But constraints on outputs are not the only imaginable response to the restrictiveness problem raised by the aspiration pattern. A different way for a learner to meet the challenge – one that follows the early generative notion of Morpheme Structure Constraints rather than ROTB – is to capitalize on the absence of aspiration from the lexicon in order to describe the lexicon more succinctly. If aspiration can be squeezed out of the inventory of primitives from which underlying material is chosen, each choice in the lexicon would cost fewer bits of information. Grammars that ban underlying aspiration will thus rule out URs like $/at^h$/ and $/kik^ht/$ and, consequently, will block surface aspiration in all inappropriate contexts. Similar considerations of economy have led to the idea of underspecification in phonological theory (see Archangeli, 1988 for an elaboration of this connection, and see Steriade, 1995 for much relevant discussion), and the feature-based encoding of the lexicon that we made use of so far fits in naturally with this line of reasoning.

At this stage we will not attempt to incorporate a mechanism of feature underspecification into our OT system. Instead, we will explore a segment-based parallel of the same idea that will allow us to keep our representations simple: aspiration will be represented as an individual segment $[^h]$, allowing the learner to minimize description length by removing instances of $[^h]$ from the lexicon. The lexicon will include a dynamic inventory of segments (initially identical to the set of segments made available by the feature table), whose length will be measured as well: removing aspiration from this inventory, thus banning aspiration from URs, will shorten the encoding of the lexicon. Formally, the lexicon in (19a) is transformed into the string in (19b), with a delimiter separating the inventory from the URs. The segments in the fixed feature table provided initially to the learner, in addition to the delimiter, are enumerated and assigned a fixed binary code. The procedure is identical to the one described in section 3.1. Choices of segments for describing the lexicon are made from the new inventory, not from the original feature table. Accordingly, describing each lexical segment costs $\lceil \lg n + 1 \rceil$ bits of information, where n is the number of segments in the new inventory, and 1 is added due to the presence of the delimiter symbol. Measuring the size of the constraint set and the size of $D{:}G$ remains the same.[19] We will now present the learning setting and show that this solution leads to correct predictions.

(19) a. $\{k^hat, ip, k^hatpit\}$

b. $\underbrace{{}^haikpt\#}_{inventory}\underbrace{k^hat\#ip\#k^hatpit\#}_{lexicon}$

The alphabet for our pseudo-English case was $\Sigma = \{a, i, u, p, t, k, ^h\}$, and we used the feature table in (20). We generated all monosyllabic words of the form $\{CV, VC, CVC\}$ and all bisyllabic words of the form $\{CVCV, VCVC, CVCCV, CVCVC, CVVC\}$ over Σ (excluding $[^h]$; $n = 774$). We then randomly selected 200 words, in which we inserted aspiration after every stop that preceded a vowel.

(20)

	cons	*stop*	*spread glottis*	*velar*	*labial*	*high*
a	-	-	-	-	-	-
i	-	-	-	-	-	+
u	-	-	-	-	+	+
p	+	+	-	-	+	-
t	+	+	-	-	-	-
k	+	+	-	+	-	+
^{h}	+	-	+	-	-	-

As before, the initial state had one constraint FAITH and a lexicon identical to the data. Note that the segmental inventory is now specified next to the lexicon:

(21) Initial grammar:

a.

$$G_{initial} = \begin{cases} \text{LEX:} & \{a, i, u, p, t, k, ^h\}; up, t^hi, k^hat, ip^huk, p^hikp^hu, t^hik^hut\dots \\ \text{CON:} & \text{FAITH} \end{cases}$$

$$\text{Description length: } |G_{initial}| + |D{:}G_{initial}| = 4,359 + 160,000 = 164,359$$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & \{a, i, u, p, t, k\}; up, ti, kat, ipuk, pikpu, tikut \ldots \\ \textsc{Con}: & *[+stop][-cons] \gg \textsc{Faith} \gg \textsc{Max}([-spread\ glottis]) \end{cases}$$

$$\text{Description length: } |G_{final}| + |D{:}G_{final}| = 3,402 + 160,000 = 163,402$$

The final grammar includes a markedness constraint that militates against sequences of a stop followed by a vowel ($*[+stop][-cons]$) and a $\textsc{Max}([-spread\ glottis])$[20] constraint that ensures that aspiration is the only possible repair. Aspiration is entirely removed from the lexicon and inserted by the grammar in the right context. In the tableau presented below, candidates (a), (c), and (d) demonstrate the role played by the markedness constraint, while candidates (e) and (f) show the significance of the learned MAX constraint in preventing overgeneration.

(22)

	/kat/	$*[+stop][-cons]$	$\textsc{Max}([-spread\ glottis])$	FAITH
a.	kat	$*!$		
b. ☞	k^{h}at			$*$
c.	ktat	$*!$		$*$
d.	kiat	$*!$		$*$
e.	at		$*!$	$*$
f.	kt		$*!$	$*$
g.	k^{h}at^h			$**!$
	...			

The segmental inventory has been restricted to $\{a, i, u, p, t, k\}$, blocking aspiration in other contexts as expected; since aspiration cannot be used to describe underlying segments, no UR can derive forms like $^*at^h$ and $^*k^hik^ht$. In the examples provided to the learner as part of the simulation, aspiration of p in ip and of k in p^hikp^hu would be ungrammatical. The allophonic distribution has been learned as expected.

3.4 Optionality

The tension between economy and restrictiveness becomes particularly clear in cases that involve optional phonological processes. The significance of optionality to learnability was articulated by Baker (1979) and Dell (1981), who noted that optionality leads an economy-only evaluation metric, such as that provided in SPE, directly into the subset problem. In this section we present a learning simulation modeled after one of Dell's cases and demonstrate how MDL provides the desired remedy when optionality is concerned.

Let us first consider a concrete example, a modified version of one of Dell's French examples, before moving on to state the problem more generally. Dell's original example concerns the optional deletion of l word finally when preceded by an obstruent but not when preceded by a sonorant (and when followed by a pause or a consonant-initial word). Thus, *table* 'table' can be pronounced [tabl] or [tab] in the appropriate context, while *parle* 'speak' is always pronounced [parl] and never *[par]. We have revised the example to allow an easy formulation of optionality in the OT framework. In OT, optionality could arise when URs have more than one optimal output. Instead of dealing with a process that optionally takes place (but might not apply), we chose to handle a case where a markedness constraint could be resolved by two distinct repairs that are equally penalized.

Consider a grammar that handles consonant clusters as follows: an unfavorable sequence C_1C_2 is optionally resolved by either i epenthesis between the two consonants or by C_2 deletion. A UR like $/tabl/$ would surface either as $[tabil]$ or as $[tab]$. In addition, the grammar generates surface forms that appear as if they could have been derived by the same process, but in fact they are not. For example, the UR $/paril/$ is faithfully mapped into $[paril]$, whereas $^*[par]$ is ungrammatical. A learner exposed to $\{[tabil]$, $[tab]$, $[paril]\}$ would face an instance of the subset problem. On the one hand, it would be justified in making the generalization that $[tabil]$ and $[tab]$ are generated from the same UR. A learning strategy based solely on economy would succeed in making this inductive leap: a grammar that includes one UR ($/tabl/$) can be more economical than a grammar that has two URs ($/tabil/$ and $/tab/$), even at the cost of introducing the relevant rule or constraint. On the other hand, if only economy is taken into consideration, a UR like $/parl/$ that is strictly simpler than an alternative $/paril/$ would be preferred. Such a grammar would correctly generate $[paril]$ from $/parl/$, but since a consonant cluster could be optionally resolved by deletion, that grammar would also generate the ill-formed $^*[par]$. The process involving optionality, which we will refer to as P, should not be extended to operating on the UR of $[paril]$. Our target grammar, G_{target}, is strictly simpler than the overly restrictive identity grammar $G_{identity}$, but it has a strictly simpler alternative, call it G_{simple}, that overgeneralizes:

(23) a. G_{simple} (economy only; overgeneralizing): Admits an overly permissive version of P.

b. G_{target} (economy and restrictiveness balanced; correct): Admits an appropriately restricted version of P.

c. $G_{identity}$ (restrictiveness only; complex grammar; under-generalizing): Does not admit P.

The problem faced by the learner, then, is to generalize beyond the data (by applying P's operation to $/tabl/$), but to prevent excessive generalization (by precluding P's operation on the UR of $[paril]$, which would generate the ungrammatical $^*[par]$).[21]

In terms of MDL, minimizing the size of the grammar would generally be beneficial unless it is counterbalanced by an increased length of data encoding given the grammar. Having to make more choices in the face of optionality results in such an increase, as we saw in section 2 for the case of *ab*-nese. In the case discussed here, the dissimilar grammatical treatment of superficially similar surface forms ($tabil$ vs. $paril$) is a consequence of differences in the compression benefits that each one provides. Encoding $[tabil]$ as the output of $/tabl/$ would require paying one bit of information to specify its choice over $[tab]$ (since this is a binary choice). Generally, collapsing $[tabil]$ and $[tab]$ into a single UR would allow enough compression to justify the cost of optionality (since the result would be at least thee segments shorter), while the slight compression gained by eliminating a single vowel i from $/paril/$ would not.

We will now show that our learner converges on the correct G_{target}, to which the MDL evaluation metric assigns the best score. Moreover, it will do so without being told which forms (if any) should be collapsed. Since our intention in this subsection is to present a proof-of-concept learning of restricted optionality, we will deviate from our earlier setting and provide the learner with the final constraint set in advance. To keep with our previous assumptions, the initial ranking of the constraints will be a faithful one. The feature table is presented in (24).

(24)

	$cons$	$high$	$stop$	son	$voice$	$labial$	$liquid$
a	-	-	-	+	+	-	-
i	-	+	-	+	+	-	-
b	+	-	+	-	+	+	-
p	+	-	+	-	-	+	-
d	+	-	+	-	+	-	-
t	+	-	+	-	-	-	-
l	+	-	-	+	+	-	+
r	+	-	-	+	+	-	-

The data consisted of three pairs ($tabil, tab, tapil, tap, labil, lab$) that were to be collapsed and two unpaired forms ($paril, radil$). Each surface form was presented 25 times to the learner.[22] In the final grammar, the learner has correctly collapsed each pair into one UR by arriving at a suitable constraint ranking. As expected, the vowel i has not been removed from the unpaired forms, despite the benefit in economy that this move could have afforded. Note that in the final grammar, the length of $D : G$ has increased significantly from that of the initial grammar (from 600 to 750); this reflects the final grammar's decreased restrictiveness due to the collapsing of pairs of forms into single URs. The increase in $|D : G|$ is more than offset by the decrease in grammar size.

(25) a. Initial grammar:

$$G_{initial} = \begin{cases} \textsc{Lex}: & tabil, tab, paril, tapil, tap, radil, labil, lab \\ \textsc{Con}: & \textsc{Faith} \gg \textsc{Dep}([-high]) \gg \textsc{Max}([-liquid]) \gg *[+cons][+cons] \end{cases}$$

Description length: $|G_{initial}| + |D{:}G_{initial}| = 589 + 600 = 1,189$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & tabl, paril, tapl, radil, labl \\ \textsc{Con}: & *[+cons][+cons] \gg \textsc{Faith} \gg \textsc{Dep}([-high]) \gg \textsc{Max}([-liquid]) \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 415 + 750 = 1,165$

The following tableau demonstrates the final grammar at work:

(26)

	/tabl/	$*[+cons][+cons]$	FAITH	DEP($[-high]$)	MAX($[-liquid]$)
a.	tabl	*!			
b. ☞	tab		*		
c.	tal		*		*!
d. ☞	tabil		*		
e.	tabal		*	*!	
	. . .				

Significantly, the overgenerating G_{simple} presented above would have led to a longer description length compared to the correct hypothesis: as shown in (27), by removing all underlying instances of i the grammar itself would have been more economical, but the overall description length would have been higher.

(27) Overgenerating grammar:

$$G_{simple} = \begin{cases} \textsc{Lex}: & tabl, parl, tapl, radl, labl \\ \textsc{Con}: & *[+cons][+cons] \gg \textsc{Faith} \gg \textsc{Dep}([-high]) \gg \textsc{Max}([-liquid]) \end{cases}$$

Description length: $|G_{simple}| + |D{:}G_{simple}| = 387 + 800 = 1,187$

	/parl/	$*[+cons][+cons]$	FAITH	DEP($[-high]$)	MAX($[-liquid]$)
a.	parl	*!			
b. ☛	par		*		
c. ☞	paril		*		
	...				

3.5 Alternations

In previous examples we considered phonological grammars that map URs to surface forms regardless of the contexts they appear in. Our next step will be to show that our learning criterion extends naturally to learning morphophonological alternations. We will examine the behavior of our learner on a dataset created by concatenating a suffix to a base set of forms. A phonological process would change some of those forms at the boundary, resulting in forms that are realized differently in two different contexts: such forms would surface faithfully when occurring independently but would be phonologically altered in the environment of the suffix. To illustrate the procedure, consider the Hebrew verbs *katav* 'write' and *daag* 'worry' along with the 2nd person feminine suffix $-t$. Assuming that Modern Hebrew speakers' obstruents assimilate in voicing to a following obstruent, our dataset would include $katav$, $kataft$, $daag$, and $daakt$. Our learner will know neither about the morphological constituency of these forms nor that pairs of them are derivationally related. Instead, we will allow the learner to perform segmentation and represent suffixes as part of the lexicon. In addition, following the lead of Goldsmith (2001), URs will be allowed to be stored with pointers to suffixes that they attach to (a pointer from a UR to a suffix means that both the UR itself and the UR combined with the suffix can be inputs to the grammar). If our view of learning as compression is correct, morphophonological alternations should fall out as by-products of two distinct mechanisms: phonological induction, which we have seen in previous sections, and segmentation, which we will now introduce. Thus, if the learner is provided with enough examples, a grammar like the following, presented schematically, should lead to a shorter description length compared to a naive grammar that memorizes the data and captures no generalizations:

(28)

$$G = \begin{cases} \textsc{Lex}: & katav_{\{-t\}}, daag_{\{-t\}}; \text{Suffixes:}\{t\} \\ \textsc{Con}: & \text{assimilation-enforcing constraint ranking} \end{cases}$$

In other words, compressing the lexicon by collapsing multiple surface forms into a single UR would justify, in terms of total description length, the addition of assimilation-enforcing constraints to Con along with their appropriate ranking. Let us see how this prediction is borne out, using a small dataset of eight words, generated according to the procedure described above. In (29) below, four basic words have been concatenated with a suffix $-t$, triggering two phonological processes. In 1-2 and 3-4, suffixation results in regressive obstruent devoicing. In 5-6, two adjacent coronals are separated by e epenthesis, thus blocking voicing assimilation. In 7-8, none of the two environments is met and the basic form remains unchanged.

(29)

1)	katav	3)	daag	5)	rakad	7)	takaf
2)	kataft	4)	daakt	6)	rakadet	8)	takaft

As in section 3.4, we provide the learner with the final constraint set in advance for the present simulation. We also do not incorporate the costs of suffixes and pointers to them: the size of the lexicon in (28) is equal to the size of the lexicon $\{katav, daag\}$; specifying a choice of a UR in order to describe a surface form is made from this latter lexicon, ignoring the cost of suffixation (e.g., unsuffixed $katav$ is specified as the UR for both its unsuffixed and suffixed outputs). See Goldsmith (2001) for much relevant discussion of how the costs of suffixes and pointers to them can be taken into account. Here, Gen is allowed to arbitrarily change segments in addition to inserting and deleting, and the search procedure is extended to perform the following additional moves in the hypothesis space: create suffix (only one suffix at a time is permitted), remove suffix, add pointer, and remove pointer. As before, a modification is chosen according to a uniform distribution over all possible changes. The feature table in (30) was provided to the learner.

(30)

	cons	*voice*	*high*	*labial*	*coronal*	ATR	*rhotic*
a	-	+	-	-	-	-	-
I	-	+	+	-	-	-	-
e	-	+	-	-	-	+	-
t	+	-	-	-	+	-	-
d	+	+	-	-	+	-	-
g	+	+	+	-	-	-	-
k	+	-	+	-	-	-	-
v	+	+	-	+	-	-	-
f	+	-	-	+	-	-	-
r	+	+	-	-	+	-	+

The learner's task in this case, then, is threefold: to discover the $-t$ suffix by performing segmentation; to learn a constraint ranking that enforces regressive devoicing and epenthesis between coronal consonants; and to collapse pairs of surface forms into a single UR, without knowing in advance which forms should be collapsed. In the results presented below, all three goals have been reached. Note that the grammar includes the markedness constraints $*\begin{bmatrix} +cons \\ +voice \end{bmatrix}[-voice]$ and $*[+coronal][+coronal]$ that trigger voicing and epenthesis respectively. All other constraints, when appropriately ranked, enable the elimination of losing candidates.

(31) a. Initial grammar:

$$G_{initial} = \begin{cases} \textsc{Lex}: & katav, daag, rakad, takaf, kataft, daakt, rakadet, \\ & takaft; \text{Suffixes:}\{\} \\ \textsc{Con}: & \textsc{Faith} \gg \textsc{Max}([+cons]) \gg \textsc{Dep}([-\text{ATR}]) \gg \textsc{Ident}([-voice]) \gg \\ & \gg \textsc{Ident}([+cons]) \gg \textsc{Ident}([+labial]) \gg \textsc{Ident}([-labial]) \gg \\ & \gg \textsc{Ident}([-high]) \gg \textsc{Ident}([+high]) \gg *[-coronal][+\text{ATR}] \gg \\ & \gg *[+coronal][+coronal] \gg * \begin{bmatrix} +cons \\ +voice \end{bmatrix} [-voice] \end{cases}$$

Description length: $|G_{initial}| + |D{:}G_{initial}| = 864 + 24 = 888$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & katav_{\{-t\}}, daag_{\{-t\}}, rakad_{\{-t\}}, takaf_{\{-t\}}; \text{Suffixes:}\{t\} \\ \textsc{Con}: & * \begin{bmatrix} +cons \\ +voice \end{bmatrix} [-voice] \gg *[-coronal][+\text{ATR}] \gg \\ & \gg *[+coronal][+coronal] \gg \textsc{Ident}([-high]) \gg \textsc{Ident}([-voice]) \gg \\ & \textsc{Dep}([-\text{ATR}]) \gg \textsc{Faith} \gg \textsc{Ident}([+labial]) \gg \textsc{Max}([+cons]) \gg \\ & \textsc{Ident}([-labial]) \gg \textsc{Ident}([+cons]) \gg \textsc{Ident}([+high]) \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 520 + 16 = 536$

4 Previous Proposals

In the present section we will use the perspective provided by the evaluation metric to take a critical look at previous learning algorithms that have been proposed in the literature on OT. We start, in section 4.1, by briefly reviewing the main efforts in the literature, efforts that, as we will explain, have a somewhat different focus from our own. The next two sections concern proposals that are much closer to ours: Maximum-Likelihood Learning of Lexicons and Grammars (Jarosz, 2006b,a, 2010), discussed in section 4.2; and Lexical-Entropy Learner (Riggle, 2006), discussed in section 4.3. We will see that each proposal targets one of the two criteria of economy and restrictiveness but not both, leading to challenges of the kind discussed above for the scientist. Our own proposal, presented earlier, can thus be seen as subsuming both, balancing in a principled way between the two biases.

4.1 Constraint Re-ranking Approaches

As mentioned in the introduction, the literature on OT has taken considerations of learning very seriously. Obviously, we will not be able to do justice to all the relevant literature within the scope of this paper. For the most part, however, this literature has concerned itself with questions that are quite different from those motivating the present proposal. Specifically, some of the most influential proposals, such as Recursive Constraint Demotion (RCD; Tesar and Smolensky, 1998, 2000), Biased Constraint Demotion (BCD; Prince and Tesar, 2004), the Gradual Learning Algorithm (GLA; Boersma and Hayes, 2001), and the Maximum Entropy model of Goldwater and Johnson (2003), assume that the learner has access to pairs of URs and surface forms (as well as a finite inventory of universal constraints). Clearly, these works do not suppose that the child is given these pairs explicitly by the environment. Rather, we are to think of such proposals as part of a bigger system that includes also a learner for the pairings of URs and surface forms. Since it is integrated learners for both constraint rankings and lexicons that we are interested in, we hope that for the time being it is reasonable to set aside proposals of this kind that rely on an unspecified learner to obtain pairings of URs and surface forms.

Among constraint re-ranking approaches, there is one family of proposals, which we will refer to as paradigm-based lexicon learners, in which constraint re-ranking is combined with some lexical learning. These proposals, which include Tesar (2006, 2009, 2014), Apoussidou (2007), Merchant (2008), and Akers (2012), have the following in common: they all use paradigms to extract information about alternations, which in turn supports the learning of properties of URs. Consider, for example, a language like German in which a voicing contrast in obstruents is neutralized word finally. Given a pairing of paradigmatically related surface forms such as [rat] 'wheel.SG' and [redɐ] 'wheel.PL', paradigm-based learners may conclude that the UR in both cases is /rad/; in particular, the UR is non-identical to the surface form [rat]. Outside of alternations, paradigm-based learners posit URs that are identical to the surface forms, thus following the principle of Lexicon Optimization (Prince and Smolensky, 1993; Inkelas, 1995).

Alternations are a central source of information, and we agree with the paradigm-based approach that this source should not be overlooked. For example, it is hard to think of a different basis for learning that the UR for 'wheel' in German is /rad/ while that of 'council, advice' is /rat/: the surface form in both cases is [rat]; but the plural form of 'wheel' is [redɐ], while that of 'council' is [retə]. However, while alternations are undoubtedly important in discovering URs, they are a special case of a more general phenomenon and would ideally fall out of whatever mechanism handles the induction of phonological patterns and of the lexicon. The MDL learner that we presented above treats alternation-based learning as exactly this kind of special case, as we have tried to show in section 3.5. Paradigm-based learners, on the other hand, treat alternations as a world unto itself. Not surprisingly, then, the paradigm-based learners in the literature offer no obvious generalization for properties of URs that do not involve alternations.

The challenge for constraint re-ranking and Lexicon Optimization has been discussed by Alderete and Tesar (2002), McCarthy (2005), and Krämer (2012), who show that constraint re-ranking learners – whether paradigm-based or not – must be modified so as to learn non-identical mappings from surface forms to non-alternating URs. McCarthy discusses evidence from languages like Choctaw, Japanese, Rotuman, and Sanskrit, in which some non-alternating URs are claimed to be distinct from their surface forms. McCarthy suggests that, in these languages, non-identical mappings in alternating forms are extended to non-alternating forms. Krämer (2012) discusses this and other ways in which non-identical mappings can be inferred for non-alternating forms.

At present, however, these ideas have yet to be turned into explicit learners, leaving the task of learning non-identical mappings for non-alternating forms as a challenge for constraint re-ranking approaches, including paradigm-based learners.

But regardless of whether or not it turns out to be feasible to use information from alternations to infer non-identical mappings in non-alternating forms, Alderete and Tesar (2002) make an even stronger claim: that non-trivial learning must take place even in the absence of alternations. The argument is based on stress-epenthesis interactions in languages like Yimas, Mohawk, and Selayarese. For a particularly transparent example of learning without alternations recall the case of *ab*-nese and in particular the discussion of how the constraint **bb* might interact with a hypothetical pattern, considered in section 2.1 above, of lengthening of the penultimate segment words. We noted that a surface form such as $[aab : ab]$, with penultimate lengthening, would provide support to $/aabb/$ as the corresponding UR. There was no alternation involved to help with this inference, and none was needed. A real-world counterpart of the artificial *ab*-nese example is the interaction of stress and epenthesis in Yimas, used by Alderete and Tesar (2002) to argue explicitly for learning non-identical mappings from surface forms to URs even in the absence of supporting alternations. Until a paradigm-based learner is proposed that generalizes beyond alternations, we conclude that, like the constraint re-ranking approaches mentioned above, such learners can be set aside within the context of the present discussion. We now turn to two learners that, contrasting with constraint re-ranking approaches, support the learning of non-identical mappings also for non-alternating forms, making them more directly comparable to our MDL-based learner.

4.2 Maximum-Likelihood Learning of Lexicons and Grammars

Jarosz (2006b,a) proposes an algorithm, Maximum-Likelihood Learning of Lexicons and Grammars (MLG), for learning lexicons and constraint rankings based on the principle of Maximum Likelihood (ML). Working within a probabilistic version of OT, Jarosz assumes that a hypothesis is a distribution over constraint rankings coupled with a distribution over URs for each morpheme.[23] The learner is given the set of constraints in advance (either as part of the innate component or perhaps through a separate module for learning constraints), along with candidate URs for each morpheme. The learner then attempts to find the hypothesis that maximizes the likelihood of the data. The search starts with an uncommitted lexicon, in which all candidates for any given morpheme are equally likely, and the search for the best hypothesis is performed by the Expectation Maximization algorithm (EM; Dempster et al., 1977).

Let us demonstrate with a simple variation on *ab*-nese in which we have the same data sequence as in (1) above but are restricted to working with the constraints **ab*, **p*, and IDENT, all three of which are given to us in advance; we will also assume the knowledge that *b* has *p* as a featural variant and that *a* has *e*. The learning process will start with the hypothesis that for any given morpheme, all possible URs are equally likely. That is, the initial hypothesis provides the following distribution over the lexicon (along with a distribution over the possible rankings of the constraints):

(32) a. M_1 =(ab) URs: *ab* (.25); *ap* (.25); *eb* (.25); *ep* (.25)

b. M_2 =(bab) URs: *bab* (.125); *bap* (.125); *beb* (.125); *pab* (.125); *bep* (.125); *pap* (.125); *peb* (.125); *pep* (.125)

c. M_3 =(aaab) URs: *aaab* (.0625); *aaap* (.0625); *aaeb* (.0625); *aeab* (.0625); *eaab* (.0625); *aaep* (.0625); ...

d. M_4 =(aabab) URs: ...

e. M_5 =(baab) URs: ...

f. M_6 =(aababaaaabab) URs: ...

g. M_7 =(babababaa) URs: ...

h. M_8 =(babababababaabab) URs: ...

On Jarosz's assumptions, the correct morpheme for each surface form has been identified in advance and is available to the learner. Using this knowledge, each hypothesis defines a probability distribution over surface forms that can be computed by enumerating the possible URs and the different constraint rankings. Take the surface form *ab*, for example: suppose we encounter it in a certain position in the data, and suppose further that this position has been correctly identified as expressing the morpheme M_1. Our goal is to compute its likelihood, and we do this by enumerating the different URs that M_1 is associated with – in this case, *ab*, *ap*, *eb*, and *ep* – and by computing the conditional probability of the surface form *ab* given each of the URs; the final answer is the weighted sum of these conditional probabilities, each weighted by the probability of the relevant UR:

(33) The likelihood of the surface form *ab* given that the morpheme is M_1

$$P(\text{surface} = ab|M_1) = \sum_{u \in \{ab,ap,eb,ep\}} P(\text{surface} = ab|u)P(u)$$

The probabilities of the different URs are part of each hypothesis. For example, in the initial hypothesis (32), the distribution is uniform, with each UR for M_1 having a probability of 0.25. What remains is the computation of $P(\text{surface} = ab|u)$ for any particular UR. This is done by looking at the different constraint rankings and their probabilities (again, part of every hypothesis). Let us look at how this is done for the UR *ab*:

(34)

Hypothesis H			Probability under input *ab*
Ranking r_i		$P(r_i)$	Optimal O_k
r_1	**ab* ≫ **p* ≫ IDENT	0.2	eb
r_2	**ab* ≫ IDENT ≫ **p*	0.15	eb
r_3	IDENT ≫ **ab* ≫ **p*	0.05	ab
r_4	**p* ≫ **ab* ≫ IDENT	0.1	eb
r_5	**p* ≫ IDENT ≫ **ab*	0.0	ab
r_6	IDENT ≫ **p* ≫ **ab*	0.5	ab

The probability of the surface form *ab* given the UR *ab* is obtained by summing over the rows in which the surface form *ab* is the winner. In the present case, these are the third, fifth, and sixth rows: $P(r_3) + P(r_5) + P(r_6) = 0.05 + 0.0 + 0.5 = 0.55$. By repeating the computation with the other possible URs for M_1 we obtain the required values to compute the likelihood of the surface form *ab* given M_1 according to (33).

ML addresses the restrictiveness requirement directly: any overgeneration will lead to spending probability mass on forms that do not occur.[24] An ML grammar is thus a fully restrictive one. Meanwhile, starting from an uncommitted lexicon as in (32) encourages the learner to consider hypotheses that rely on the constraints – rather than on accidents of the lexicon – to encode patterns in the input data. Such hypotheses are in line with the OT principle of ROTB. From an information-theoretic perspective, an uncommitted lexicon is one with high entropy. As we will see in the next section, lexicon entropy (though in different form from Jarosz's) can sometimes stand proxy for economy, both criteria sometimes favoring a smaller lexicon from which significant patterns have been extracted over a more complex one in which those patterns remain.

We saw in section 2.2 above that restrictiveness must be simultaneously balanced against economy in order to provide an adequate evaluation of hypotheses, and we discussed the problematic results of unchecked restrictiveness. Despite the entropic starting point, MLG suffers from the same problem. Let us start by recalling the dangers of pure restrictiveness. Without the balancing force of economy, restrictiveness will make the learner attempt a full memorization of the data. If the learner can memorize the entire sequence of data, it will do so. The only thing that can stop it is its own representational limitations. For example, if it cannot represent the order of elements in the input data, it will have to resort to an approximation.[25] In our modified *ab*-nese case, the following hypothesis will receive likelihood 1, the highest possible score:

(35) a. $M_1 =$(ab) URs: *ab* (1); *ap* (0); *eb* (0); *ep* (0)

b. $M_2 =$(bab) URs: *bab* (1); *bap* (0); *beb* (0); *pab* (0); *bep* (0); *pap* (0); *peb* (0); *pep* (0)

c. $M_3 =$(aaab) URs: *aaab* (1); *aaap* (0); *aaeb* (0); *aeab* (0); *eaab* (0); *aaep* (0); . . .

d. $M_4 =$(aabab) URs: *aabab* (1) . . .

e. $M_5 =$(baab) URs: *baab* (1) . . .

f. $M_6 =$(aababaaaabab) URs: *aababaaaabab* (1) . . .

g. $M_7 =$(babababaa) URs: *babababaa* (1) . . .

h. M_8 =(babababababaabab) URs: *babababababaabab* (1) …

(36) IDENT $\gg$ **ab* $\gg$ **p*

The hypothesis summarized in (35) and (36) is clearly not what we want: it has simply memorized the data, leaving the absence of *p* as an accident of the (uncompressed) lexicon. By Jarosz's ML criterion, however, this hypothesis obtains a perfect score; other contenders can at most obtain a tie with this memorized hypothesis. This problem is quite general: as long as we can list the observed surface forms as having probability 1 (and as long as we can rely on morpheme identification, as in Jarosz's examples), MLG will give likelihood 1 to the fully memorized hypothesis, using the ranking of faithfulness over markedness. Going back to our original *ab*-nese case, ML will see no benefit in squeezing **bb* out of the lexicon and into the constraints. In English, the same will hold with respect to aspiration: if IDENT outranks the other constraints, a lexicon that memorizes the surface forms with probability 1, including aspiration, will give the data likelihood 1 (again, the highest possible score). We will thus be left without an account of why speakers of English fail to notice the difference between the aspirated t^h in 'tack' and the unaspirated *t* in 'stack'.[26] Worse, there will never be any generalization. As we discussed in section 2.2 above, restrictiveness alone will fail on any input sequence that shows a proper subset of the possible forms. Earlier we demonstrated this for the phonologist in the case of *zab*-nese, in which any nonnegative number of *z*'s can precede any word. As we discussed, a restrictiveness-only phonologist will fail to make this generalization, memorizing instead the finite subset of the infinite allowable *z*-forms seen so far and assigning zero probability to any of the accidental gaps in the input data. MLG, similarly aiming for restrictiveness only, will run into the very same problem.

What about the uncommitted initial state as a cure for memorization? In Jarosz's examples, the learner does not end up memorizing the input data, and we mentioned that the uncommitted initial state of MLG is designed to encourage the learner to be reasonable. Unfortunately, such encouragement is generally short lived. It affects the beginning of the search, but if the search is capable enough, the ML criterion will necessarily lead the learner to a maximally memorized hypothesis. That Jarosz's examples do not exhibit such memorization we must attribute to peculiarities of the search procedure. The EM algorithm is known to get caught in local optima, which could account for these results. Moreover, it is possible that the search has stopped before convergence. In other words, what prevents the problematic ML metric from hurting the learning process is a problematic search procedure that fails to find hypotheses that are global winners in terms of ML. Since modeling the search goes beyond the goals of current research, we conclude that the entropic initial state is not capable of rescuing ML as the learning criterion for the child.

4.3 Lexical-Entropy Learner

We just saw that an uncommitted – or entropic – initial state does little to help the learner. Assuming that an entropic lexicon is indeed a relevant property of good hypotheses, the remedy seems clear: turn the requirement into an active force by incorporating it into the learning criterion. This is exactly what Riggle (2006) proposes. On Riggle's account, different grammar and lexicon hypotheses are evaluated according to a measure of lexicon entropy. The measure, which is somewhat different from Jarosz's and which we will discuss shortly, is based on the following principle:

(37) Lexicon entropy principle: Assume a universal constraint set CON. Whenever faced with a decision as to whether to encode a phonological pattern as a consequence of constraint interaction or as an accident of the lexicon, the former option must be taken. (Modified from Riggle, 2006, p. 347.)

Riggle proposes the evaluation of a grammar G according to the conditional entropy of G's lexicon, defined in terms of bigrams as follows:

(38)

$$H(G) = -\sum_{x\in\Sigma}\sum_{y\in\Sigma} P(x,y)\log P(y|x)$$

Given two hypotheses consistent with the data, the learner is expected to prefer the one for which $H(G)$ is higher. As an example of how this should work, consider again the *ab*-nese data from section 2.1, and the three constraints **bb*, MAX, and DEP. We will show why Riggle's metric rejects the identity hypothesis in favor of the correct lexicon and ranking combination. The data are repeated here, along with the two competing hypotheses:[27]

(39) Hypothesis 1 (identity)

Lexicon:

1) /ab/ 3) /aaab/ 5) /abaab/ 7) /babababaa/

2) /bab/ 4) /aabab/ 6) /babaaaa/ 8) /bababababaabab/

Corresponding ranking: any

Entropy: 0.57

(40) Hypothesis 2 (correct)

Lexicon:

1) /ab/ 3) /aaab/ 5) /abaab/ 7) /bbbbaa/

2) /bb/ 4) /aabb/ 6) /bbaaaa/ 8) /bbbbbbaabb/

Corresponding ranking: **bb*, Max $\gg$ Dep

Entropy: 0.88

The lexicon of hypothesis 1 is identical to the surface data. Under any ranking of the three constraints, all underlying representations would surface unchanged. The generalization that a sequence bb is prohibited in *ab*-nese is captured only as an accident of the lexicon. As a consequence, the lexicon contains predictable information that can be identified by computing probabilities of adjacent segments: after seeing a b, there is a probability of 1.0 that a following segment be a. Formally, $-P(b,b)\log P(b|b)$ and $-P(b,a)\log P(a|b)$ will both be null (assuming here for simplicity's sake that $0\log 0 = \lim_{x\to 0} x\log x = 0$), not adding to the entropy of the lexicon, which results in 0.57.

On the other hand, the second hypothesis has the predictable information about the absence of *bb* sequences removed from the lexicon, resulting in a more irregular lexicon: seeing a consonant or a vowel, it is hard to predict what the next segment would be. Here, all summands contribute to the measure of entropy, which sums to 0.88 — a higher entropy than that of the identity hypothesis. Importantly, given the lexicon of the second hypothesis, a sequence bb must be resolved by vowel epenthesis, entailing the more restrictive ranking **bb* $\gg$ Dep.

We can see that Riggle uses entropy as a proxy for economy. In his proposal, entropy is the only factor in the learning criterion. In particular, there is no pressure for restrictiveness. This choice leads to the subset problem, the problem discussed earlier for the scientist using the original SPE evaluation metric and the mirror image of the problem for Jarosz's proposal. To see this, let us consider first a version of Riggle's proposal for *ab*-nese in which the constraints are not given in advance and must be learned. In the absence of a pressure for restrictiveness, an entropic but overgenerating grammar like the following will fare much better than the correct grammar:

(41) Hypothesis 3 (no constraints; entropic; overgenerates)

Lexicon: /aabba/

Corresponding ranking: (NONE: no constraints to rank)

Entropy: 1.0

In Lexicon 3, the bigram distribution is uniform: $P(x|y)$ is the same (= 0.5) for any x and y. It is thus maximally entropic. At the same time, it massively overgenerates: in the absence of any constraints, the single UR */aabba/* can be mapped to any of the attested forms but also to any other form, all without incurring a single violation. In this case, then, entropy alone exposes the learner to the subset problem, just as economy alone exposed the scientist to this problem in our discussion earlier.

In Riggle's actual proposal, the constraints are given to the learner in advance. With a judicious choice of constraints, the subset problem is ameliorated, but we will show that it does not disappear completely. Let us continue with our *ab*-nese example, and let us assume that the learner is given the set of constraints that our phonologist from section 2 arrived at. In this case, Lexicon 3 is no longer appropriate (since it does not generate the data), but the following overgenerating hypothesis is just as entropic as the intuitively correct Hypothesis 2:

(42) Hypothesis 4 (overgenerates)

Lexicon:

1) /ab/ 3) /aaab/ 5) /abaab/ 7) /bbbbaa/

2) /bb/ 4) /aabb/ 6) /bbaaaa/ 8) /bbbbbb/

Corresponding ranking: *bb* $\gg$ Max, Dep

Entropy: 0.88

Hypothesis 4 keeps the ranking *bb* $\gg$ Dep, but it has Max ranked together with Dep rather than above it. As a result, all the correct surface forms are still generated from the intuitively correct lexicon, but along with them we will also find unattested forms such as *b* (from the UR *bb*), generated through *b*-deletion. This is an overgeneration problem: as long as URs with the sequence *bb* are chosen with nonzero probability, the hypothesis wastes probability mass on forms such as *b* that will never actually occur. Since lexicon entropy does not take restrictiveness into account, such overgeneration will not lead to Hypothesis 4 being dispreferred.[28]

In order to assess the suitability of entropy as a pressure on hypotheses, then, we must combine it with a pressure for restrictiveness. A natural way to accomplish this is by combining it with Jarosz's ML criterion. There are many different ways to combine two criteria into one, and many of these (such as maximizing the sum – or the product – of the likelihood of the data and the entropy of the lexicon) will address the problem of overgeneration without degenerating into memorizing the input data.

Unfortunately, no combination of Riggle-like lexicon entropy with ML can work. To see why, consider again the two lexicons for *ab*-nese that seemed to justify the entropy criterion. Lexicon 1 was more complex and less entropic than Lexicon 2, which seemed encouraging. But consider now Lexicon 5, a lexicon based on *c*-deletion rather than *a*-insertion:

(43) Hypothesis 5 (entropic and restrictive but presumably bad)

Lexicon:

1) /cacbc/
2) /bab/
3) /caaaccb/
4) /aabab/
5) /abaab/
6) /babaaaa/
7) /babababaa/
8) /babababababaabab/

Corresponding ranking: *c, *bb, Dep $\gg$ Max

Entropy: 0.95

Lexicon 5 is more complex still than Lexicon 1 but it is more entropic than either Lexicon 1 or Lexicon 2. In fact, infinitely many such lexicons are easily constructed, each more pointlessly complex than the other and with higher entropy. Note that all the hypotheses in this case are fully restrictive, so ML will not help choose between them. The decision is down to entropy, and entropy leads us astray: it only cares about making the grammar less regular, but this can be accomplished not just by removing orderly material, which is what we would like, but also by adding disorderly material, which we would not. We conclude that economy must be represented directly, as it is under MDL, rather than by proxy.

5 Discussion

5.1 MDL as Guide for Learning

The simulation results address concerns sometimes raised in the literature regarding the ability of MDL to yield the right results for learning. In particular, Adriaans and Jacobs (2006) and Adriaans (2007) analyze the effects of MDL as a guide to learning and reach ambivalent conclusions, focusing on the induction of deterministic Finite-State Automata (DFAs). They show, using a measure of goodness that they refer to as *randomness deficiency*, that between two given DFAs, the one with the lower description length is not necessarily the one that fares better with respect to randomness deficiency; the DFA that minimizes description length globally, however, is best also in terms of randomness deficiency. They conclude that MDL is a good guide globally but a poor one locally.

In our simulations, progress is made by local comparisons of description length, which in principle could lead to the kind of problem noted by Adriaans and Jacobs. (The constraints and lexicons in our representations are quite different from Adriaans and Jacobs's DFAs, but the challenge they raise is presumably quite general.) However, our search uses Simulated Annealing, which, as discussed above, can escape local traps by switching from a good hypothesis to a worse one from time to time. As the simulation results show, our search indeed manages to reach the target grammars in due course.

Adriaans and Jacobs (2006) raise another challenge to MDL learners: they observe, again in the context of DFA induction, that such learners can be extremely sensitive to the choices that are made in the encoding schemes. Here we are less sure what to say. On the one hand, our simulations show that MDL supports successful learning within a linguistically significant formalism and across several different patterns. On the other hand, we have not undertaken a study of the robustness of the learner to different choices. In our simulations, we have used various parameter values that seemed sensible: most significantly, a certain initial temperature, a cooling agenda, and a multiplier for the data. Anecdotally, different choices did not seem to make much of a difference. Clearly, though, a systematic survey is in order.

5.2 MDL and the Typology

Heinz and Idsardi (2013) discuss a typological challenge for learning theories in phonology: most attested phonological patterns are captured by particular subclasses of finite-state automata but not by others. This does not appear to be an accident, which raises the question of how it can be accounted for. Heinz and Idsardi propose an account in terms of learning procedures that work with the attested subclasses of automata but do not cover the unattested ones. Within this context, they express skepticism about a role for MDL in addressing this challenge based on the following observation: sometimes a finite-state automaton that captures a typologically attested kind of pattern is bigger than an automaton that captures a typologically unattested kind of pattern. If the unattested patterns are representable, and if MDL is stated over finite-state automata, then MDL is unlikely to bias learners towards the attested patterns.

Heinz and Idsardi discuss the length of very specific representations – namely, the finite-state machines they use to describe the relevant patterns – and these representations do not correspond to any of the main grammatical formalisms for phonology. (In particular, as we discussed in detail above, the representations we have been using are quite different.) Given different representations, grammar size can change, and we do not know whether Heinz and Idsardi's observation carries over to OT. But suppose, for the purpose of the present discussion, that it did. As far we are aware, none of the previous proposals in the OT learnability literature (including those discussed above) attempts to account for typological patterns through the learning algorithm: a common assumption within OT is that the learning mechanism must be able to induce the correct grammar from a sufficient amount of typical data. Typological patterns, on this view, may arise through the formalism – specifically, certain patterns will not be representable – as well as through various factors having to do with communication, error, and exposure time. Critically, though, they are not due to the inherent inability of the learner to learn something that can be represented. Given this common assumption, a typological pattern such as Heinz and Idsardi's is generally no more challenging for one theory of OT learning than it is for another. In the present paper we have adopted this common assumption. Our goal has been to argue for MDL as a learning theory for OT, and Heinz and Idsardi's generalization is currently of no help in choosing between such theories.

There is one typological question that the present work raises and that we will leave open. In the literature on OT, much of the burden of accounting for the factorial typology falls on the innate set of constraints. To be sure, the influence of factors such as communication pressure is not denied, but a great many number of generalizations about markedness are commonly taken to arise from innately provided constraints that the learner brings to the task. The present paper does not directly challenge this assumption. Indeed, we have shown that the MDL evaluation metric can yield successful results in two cases – French-like optionality and Hebrew-like alternations – in which the constraints were given in advance. But we have also shown that the MDL evaluation metric can succeed in two cases – *ab*-nese and English-like aspiration – in which the constraints were not given in advance and had to be induced from general constraint schemata. The simulations involving constraint induction can be seen as motivation to investigate variants of OT in which at least some of the constraints are learned, but any variant of this kind will need to account for the typological patterns that those constraints were meant to capture.

5.3 Learning Across Components

We saw how the generality of the MDL evaluation metric allows it to apply to alternations without changing the metric itself: all that was needed was an enrichment of the representations to include possible suffixation. More broadly, we can consider learning across components, specifying the representations for phonetics and morphology, for example, and letting the MDL evaluation metric lead us to predictions regarding the order in which patterns in the different components are learned. This is a natural continuation of the current work and one that we find interesting. We note, however, that this direction – in which the learning criterion is uniform and any differences in learning between components derives from differences in the possible representations – seems incompatible with the view articulated by Heinz (2007) and Heinz and Idsardi (2013) that learning in phonology is fundamentally different from learning in other components. We hope that the present paper will make it easier to examine the two views and their implications more closely.

5.4 Comparing Architectures

Another question we wish to mention is whether predictions about learning can help choose between competing hypotheses about representations. For example, we saw how the ability to restrict the alphabet used within the lexicon gives the MDL learner a handle on learning the distribution of aspiration in English. The status of such constraints on URs has been the topic of debate in OT, with most of the literature rejecting constraints of this kind. The MDL evaluation metric holds the promise of providing such architectural choices with an interpretation in terms of predictions about learning.

6 Conclusion

We have argued for the MDL evaluation metric as the criterion for hypothesis comparison by the learner. At first glance, the compression criterion at the heart of Solomonoff's metric can seem foreign from the perspective of OT. We tried to show, however, that this criterion is in fact familiar from the everyday work of the phonologist. We then presented the case for using this criterion as a methodologically natural starting point to study the child's learning criterion: given any theory of UG, the ability to store grammars in memory and use them to parse the data already provides the basis for using the MDL metric. We proceeded to present several simulation results showing how the MDL metric can be used by a learner trying to make sense of raw data. While clearly preliminary, these proof-of-concept results – all of them new – show how lexicons and constraint rankings can be induced, with and without supporting data from alternations, and how the same metric extends to learning the constraints themselves.

Appendix A: Data

ab-nese

a, ab, ba, bab, aa, aab, aba, abab, baa, baab, baba, babab, aaa, aaab, aaba, aabab, abaa, abaab, ababa, ababab, baaa, baaab, baaba, baabab, babaa, babaab, bababa, babababab, aaaa, aaaab, aaaba, aaabab, aabaa, aabaab, aababa, aababab, abaaa, abaaab, abaaba, abaabab, ababaa, ababaab, abababa, abababab, baaaa, baaaab, baaaba, baaabab, baabaa, baabaab, baababa, baababab, babaaa, babaaab, babaaba, babaabab, bababaa, bababaab, babababa, babababab, aaaaa, aaaaab, aaaaba, aaaabab, aaabaa, aaabaab, aaababa, aaababab, aabaaa, aabaaab, aabaaba, aabaabab, aababaa, aababaab, aabababa, aabababab, abaaaa, abaaaab, abaaaba, abaaabab, abaabaa, abaabaab, abaababa, abaababab, ababaaa, ababaaab, ababaaba, ababaabab, abababaa, abababaab, ababababa, ababababab, baaaaa, baaaaab, baaaaba, baaaabab, baaabaa, baaabaab, baaababa, baaababab, baabaaa, baabaaab, baabaaba, baabaabab, baababaa, baababaab, baabababa, baabababab, babaaaa, babaaaab, babaaaba, babaaabab, babaabaa, babaabaab, babaababa, babaababab, bababaaa, bababaaab, bababaaba, bababaabab, babababaa, babababaab, bababababa, bababababab, aaaaaa, aaaaaab, aaaaaba, aaaaabab, aaaabaa, aaaabaab, aaaababa, aaaababab, aaabaaa, aaabaaab, aaabaaba, aaabaabab, aaababaa, aaababaab, aaabababa, aaabababab, aabaaaa, aabaaaab, aabaaaba, aabaaabab, aabaabaa, aabaabaab, aabaababa, aabaababab, aababaaa, aababaaab, aababaaba, aababaabab, aabababaa, aabababaab, aababababa, aababababab, abaaaaa, abaaaaab, abaaaaba, abaaaabab, abaaabaa, abaaabaab, abaaababa, abaaababab, abaabaaa, abaabaaab, abaabaaba, abaabaabab, abaababaa, abaababaab, abaabababa, abaabababab, ababaaaa, ababaaaab, ababaaaba, ababaaabab, ababaabaa, ababaabaab, ababaababa, ababaababab, abababaaa, abababaaab, abababaaba, abababaabab, ababababaa, ababababaab, abababababa, abababababab, baaaaaa, baaaaaab, baaaaaba, baaaaabab, baaaabaa, baaaabaab, baaaababa, baaaababab, baaabaaa, baaabaaab, baaabaaba, baaabaabab, baaababaa, baaababaab, baaabababa, baaabababab, baabaaaa, baabaaaab, baabaaaba, baabaaabab, baabaabaa, baabaabaab, baabaababa, baabaababab, baababaaa, baababaaab, baababaaba, baababaabab, baabababaa, baabababaab, baababababa, baababababab, babaaaaa, babaaaaab, babaaaaba, babaaaabab, babaaabaa, babaaabaab, babaaababa, babaaababab, babaabaaa, babaabaaab, babaabaaba, babaabaabab, babaababaa, babaababaab, babaabababa, babaabababab, bababaaaa, bababaaaab, bababaaaba, bababaaabab, bababaabaa, bababaabaab, bababaababa, bababaababab, babababaaa, babababaaab, babababaaba, babababaabab, bababababaa, bababababaab, babababababa, babababababab

Aspiration

ak, ap, it, up, k^{h}u, p^{h}i, p^{h}u, t^{h}i, k^{h}at, k^{h}it, k^{h}up, p^{h}it, p^{h}up, t^{h}ik, t^{h}ip, t^{h}it, ak^{h}ap, ak^{h}ik, ap^{h}ap, ap^{h}at, ap^{h}ik, ap^{h}ip, at^{h}it, at^{h}uk, ik^{h}ak, ip^{h}ap, ip^{h}ik, ip^{h}ip, ip^{h}uk, ip^{h}up, it^{h}ap, it^{h}it, k^{h}aat, k^{h}aik, k^{h}ait, k^{h}auk, k^{h}aup, k^{h}iap, k^{h}iat, k^{h}iuk, k^{h}uap, p^{h}aik, p^{h}ait, p^{h}aut, p^{h}iik, p^{h}iit, p^{h}uat, p^{h}uik, p^{h}uit, t^{h}ait, t^{h}auk, t^{h}iup, uk^{h}ak, uk^{h}ut, up^{h}ik, up^{h}ip, up^{h}ut, ut^{h}up, k^{h}ak^{h}u, k^{h}ip^{h}a, k^{h}uk^{h}u, p^{h}ak^{h}i, p^{h}ak^{h}u, p^{h}ap^{h}u, p^{h}at^{h}a, p^{h}ik^{h}u, p^{h}uk^{h}u, p^{h}up^{h}u, t^{h}ak^{h}i, t^{h}at^{h}a, t^{h}at^{h}u, t^{h}ik^{h}a, t^{h}ik^{h}i, t^{h}ip^{h}u, t^{h}ut^{h}u, k^{h}ak^{h}at, k^{h}ak^{h}up, k^{h}ak^{h}ut, k^{h}akp^{h}a, k^{h}akt^{h}a, k^{h}ap^{h}uk, k^{h}ap^{h}up, k^{h}apk^{h}i, k^{h}app^{h}i, k^{h}app^{h}u, k^{h}apt^{h}i, k^{h}at^{h}ut, k^{h}atk^{h}u, k^{h}atp^{h}a, k^{h}att^{h}a, k^{h}ik^{h}ak, k^{h}ik^{h}ap, k^{h}ikp^{h}a, k^{h}ikp^{h}i, k^{h}ikt^{h}i, k^{h}ip^{h}ak, k^{h}ipt^{h}a, k^{h}ipt^{h}i, k^{h}it^{h}ap, k^{h}it^{h}at, k^{h}it^{h}ik, k^{h}it^{h}it, k^{h}itp^{h}a, k^{h}itt^{h}a, k^{h}itt^{h}i, k^{h}itt^{h}u, k^{h}uk^{h}ak, k^{h}uk^{h}ip, k^{h}uk^{h}it, k^{h}ukk^{h}a, k^{h}ukk^{h}i, k^{h}ukt^{h}a, k^{h}ukt^{h}i, k^{h}up^{h}ap, k^{h}up^{h}ik, k^{h}up^{h}ip, k^{h}up^{h}uk, k^{h}upk^{h}a, k^{h}upp^{h}a, k^{h}upp^{h}i, k^{h}upp^{h}u, k^{h}ut^{h}ak, k^{h}utp^{h}a, k^{h}utp^{h}i, k^{h}utt^{h}a, p^{h}ak^{h}ak, p^{h}ak^{h}ap, p^{h}akp^{h}i, p^{h}akp^{h}u, p^{h}ap^{h}uk, p^{h}ap^{h}up, p^{h}apk^{h}a, p^{h}apk^{h}i, p^{h}at^{h}ak, p^{h}at^{h}ik, p^{h}at^{h}it, p^{h}atp^{h}a, p^{h}ik^{h}ak, p^{h}ik^{h}uk, p^{h}ikp^{h}u, p^{h}ikt^{h}i, p^{h}ip^{h}ak, p^{h}ip^{h}uk, p^{h}ipk^{h}i, p^{h}ipp^{h}i, p^{h}it^{h}ak, p^{h}it^{h}ut, p^{h}itk^{h}u, p^{h}itp^{h}i, p^{h}uk^{h}ap, p^{h}uk^{h}up, p^{h}ukk^{h}i, p^{h}upk^{h}i, p^{h}upp^{h}a, p^{h}ut^{h}ak, p^{h}ut^{h}ap, p^{h}ut^{h}up, p^{h}utp^{h}i, p^{h}utp^{h}u, p^{h}utt^{h}i, p^{h}utt^{h}u, t^{h}ak^{h}at, t^{h}akk^{h}i, t^{h}akk^{h}u, t^{h}akp^{h}i, t^{h}akp^{h}u, t^{h}akt^{h}i, t^{h}akt^{h}u, t^{h}ap^{h}ik, t^{h}ap^{h}up, t^{h}apk^{h}u, t^{h}apt^{h}u, t^{h}at^{h}ap, t^{h}at^{h}ip, t^{h}at^{h}uk, t^{h}at^{h}ut, t^{h}atk^{h}a, t^{h}atk^{h}u, t^{h}att^{h}a, t^{h}att^{h}i, t^{h}ik^{h}uk, t^{h}ik^{h}ut, t^{h}ip^{h}ap, t^{h}ip^{h}ip, t^{h}ip^{h}it, t^{h}ipk^{h}i, t^{h}ipt^{h}a, t^{h}ipt^{h}i, t^{h}it^{h}ik, t^{h}it^{h}ut, t^{h}itk^{h}i, t^{h}itt^{h}u, t^{h}uk^{h}at, t^{h}uk^{h}ut, t^{h}ukt^{h}i, t^{h}ukt^{h}u, t^{h}up^{h}ut, t^{h}upp^{h}a, t^{h}ut^{h}it, t^{h}utk^{h}i

Optionality

tabil, tab, paril, tapil, tap, radil, labil, lab

Alternations

daag, daakt, katav, kataft, rakad, rakadet, takaf, takaft

Appendix B: Results from Segment-Based Simulations

As mentioned in section 3.3, we have used a segment-based encoding of the lexicon to test the learning of aspiration, but a feature-based encoding in all other simulations. Here we present alternative results of the three remaining simulations (ab-nese, French optionality, and Hebrew alternations) in which the segment-based encoding is used instead. The setting for each simulation is identical to the setting reported in section 3, except for the French optionality simulation, in which the input data include five words instead of eight. Otherwise, the final grammars reached are the same.

ab-nese

(44) a. Initial grammar:

$$G_{initial} = \begin{cases} \textsc{Lex}: & \{a, b\}; bab, aabab, ab, baab, babaaa, babababaa \ldots \\ \textsc{Con}: & \textsc{Faith} \end{cases}$$

Description length: $|G_{initial}| + |D{:}G_{initial}| = 4,628 + 201,600 = 206,228$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & \{a, b\}; bb, aabb, ab, baab, bbaaa, bbbbaa, \ldots \\ \textsc{Con}: & \textsc{Max}([+cons]) \gg *[+cons][+cons] \gg \textsc{Faith} \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 4,034 + 201,600 = 205,634$

Optionality

(45) a. Initial grammar:

$$G_{initial} = \begin{cases} \textsc{Lex}: & \{a, i, b, p, d, t, l, r\}; tab, tabil, tap, tapil, paril \\ \textsc{Con}: & \textsc{Faith} \gg \textsc{Dep}([-high]) \gg \textsc{Max}([-liquid]) \gg *[+cons][+cons] \end{cases}$$

Description length: $|G_{initial}| + |D{:}G_{initial}| = 208 + 375 = 583$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & \{a, i, b, p, d, t, l, r\}; tabl, tapl, paril \\ \textsc{Con}: & *[+cons][+cons] \gg \textsc{Faith} \gg \textsc{Dep}([-high]) \gg \textsc{Max}([-liquid]) \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 178 + 350 = 528$

Alternations

(46) a. Initial grammar:

$$G_{initial} = \begin{cases} \textsc{Lex}: & \{a, I, e, t, d, g, k, v, f, r\}; katav, daag, rakad, takaf, kataft, daakt, rakadet, \\ & takaft; \text{Suffixes:}\{\} \\ \textsc{Con}: & \textsc{Faith} \gg \textsc{Max}([+\mathit{cons}]) \gg \textsc{Dep}([-\text{ATR}]) \gg \textsc{Ident}([-voice]) \gg \\ & \gg \textsc{Ident}([+\mathit{cons}]) \gg \textsc{Ident}([+labial]) \gg \textsc{Ident}([-labial]) \gg \\ & \gg \textsc{Ident}([-high]) \gg \textsc{Ident}([+high]) \gg *[-coronal][+\text{ATR}] \gg \\ & \gg *[+coronal][+coronal] \gg * \begin{bmatrix} +\mathit{cons} \\ +\mathit{voice} \end{bmatrix} [-voice] \end{cases}$$

Description length: $|G_{initial}| + |D{:}G_{initial}| = 492 + 24 = 516$

b. Final grammar:

$$G_{final} = \begin{cases} \textsc{Lex}: & \{a, I, e, t, d, g, k, v, f, r\}; katav_{\{-t\}}, daag_{\{-t\}}, rakad_{\{-t\}}, takaf_{\{-t\}}; \text{Suffixes:}\{t\} \\ \textsc{Con}: & * \begin{bmatrix} +\mathit{cons} \\ +\mathit{voice} \end{bmatrix} [-voice] \gg *[-coronal][+\text{ATR}] \gg \\ & \gg *[+coronal][+coronal] \gg \textsc{Ident}([-high]) \gg \textsc{Ident}([-voice]) \gg \\ & \textsc{Dep}([-\text{ATR}]) \gg \textsc{Faith} \gg \textsc{Ident}([+labial]) \gg \textsc{Max}([+\mathit{cons}]) \gg \\ & \textsc{Ident}([-labial]) \gg \textsc{Ident}([+\mathit{cons}]) \gg \textsc{Ident}([+high]) \end{cases}$$

Description length: $|G_{final}| + |D{:}G_{final}| = 376 + 16 = 392$

References

Adriaans, Pieter. 2007. Learning as data compression. In *Computation and logic in the real world*, ed. S.Barry Cooper, Benedikt Löwe, and Andrea Sorbi, 11–24. Springer.

Adriaans, Pieter, and Ceriel Jacobs. 2006. Using MDL for grammar induction. In *Grammatical inference: Algorithms and applications*, Lecture Notes in Computer Science. Berlin: Springer.

Akers, Crystal Gayle. 2012. Commitment-based learning of hidden linguistic structures. Doctoral Dissertation, Rutgers University-Graduate School-New Brunswick.

Albright, Adam, and Bruce Hayes. 2011. Learning and learnability in phonology. In *The handbook of phonological theory*, ed. John Goldsmith, Jason Riggle, and Alan Yu, 661–690. Wiley Online Library, 2nd edition.

Alderete, John. 1999. Head dependence in stress-epenthesis interaction. In *The derivational residue in phonological optimality theory*, ed. Ben Hermans and Marc van Oostendorp, 29–50. John Benjamins.

Alderete, John, and Bruce Tesar. 2002. Learning covert phonological interaction: an analysis of the problem posed by the interaction of stress and epenthesis. Technical Report RuCCS-TR-72, Rutgers Center for Cognitive Science, Piscataway, NJ. ROA 543.

Anttila, Arto. 2007. Variation and optionality. *The Cambridge handbook of phonology* 519–536.

Apoussidou, Diana. 2007. *The learnability of metrical phonology*. LOT.

Archangeli, Diana. 1988. Aspects of underspecification theory. *Phonology* 5:183–207.

Baker, Carl L. 1979. Syntactic theory and the projection problem. *Linguistic Inquiry* 10:533–581.

Becker, Michael, Nihan Ketrez, and Andrew Nevins. 2011. The surfeit of the stimulus: Analytic biases filter lexical statistics in Turkish laryngeal alternations. *Language* 87:84–125.

Berwick, Robert C. 1982. Locality principles and the acquisition of syntactic knowledge. Doctoral Dissertation, MIT, Cambridge, MA.

Berwick, Robert C. 1985. *The acquisition of syntactic knowledge*. Cambridge, Massachusetts: MIT Press.

Boersma, Paul. 1998. *Functional phonology: Formalizing the interactions between articulatory* *and perceptual drives*. Holland Academic Graphics/IFOTT.

Boersma, Paul, and Bruce Hayes. 2001. Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32:45–86.

Braine, Martin D. S. 1971. On two types of models of the internalization of grammars. In *The ontogenesis of grammar*, ed. D. J. Slobin, 153–186. Academic Press.

Brent, Michael, and T. Cartwright. 1996. Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition* 61:93–125.

Chaitin, Gregory J. 1966. On the length of programs for computing finite binary sequences. *Journal of the ACM* 13:547–569.

Chater, Nick, and Paul Vitányi. 2007. 'Ideal learning' of natural language: Positive results about learning from positive evidence. *Journal of Mathematical Psychology* 51:135–163.

Chomsky, Noam. 1951. Morphophonemics of Modern Hebrew. Master's thesis, University of Pennsylvania.

Chomsky, Noam. 1965. *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Chomsky, Noam. 1981. *Lectures on Government and Binding*. Foris, Dordrecht.

Chomsky, Noam, and Morris Halle. 1968. *The sound pattern of English*. New York: Harper and Row Publishers.

Clark, Alexander. 2001. Unsupervised language acquisition: Theory and practice. Doctoral Dissertation, University of Sussex.

Cover, Thomas M., and Joy A. Thomas. 2006. *Elements of information theory*. Wiley-Interscience, 2nd edition.

Dell, François. 1981. On the learnability of optional phonological rules. *Linguistic Inquiry* 12:31–37.

Dempster, Arthur Pentland, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39:1–38.

Dowman, Mike. 2007. Minimum description length as a solution to the problem of generalization in syntactic theory. Ms., University of Tokyo, Under review.

Endress, Ansgar, Ghislaine Dehaene-Lambertz, and Jacques Mehler. 2007. Perceptual constraints and the learnability of simple grammars. *Cognition* 105:577–614.

Endress, Ansgar, Marina Nespor, and Jacques Mehler. 2009. Perceptual and memory constraints on language acquisition. *Trends in Cognitive Sciences* 13:348–353.

Endress, Ansgar D, and Jacques Mehler. 2010. Perceptual constraints in phonotactic learning. *Journal of experimental psychology. Human perception and performance* 36:235–250.

Gibson, Edward, and Kenneth Wexler. 1994. Triggers. *Linguistic Inquiry* 25:407–454.

Goldsmith, John. 2001. Unsupervised learning of the morphology of a natural language. *Computational Linguistics* 27:153–198.

Goldsmith, John. 2010. Towards a new empiricism for linguistics. To appear as chapter 3 in Empiricist Approaches to Language Learning, co-authored with Alex Clark, Nick Chater, and Amy Perfors.

Goldsmith, John, and Jason Riggle. 2012. Information theoretic approaches to phonological structure: the case of Finnish vowel harmony. *Natural Language & Linguistic Theory* 30:859–896.

Goldwater, Sharon, and Mark Johnson. 2003. Learning OT constraint rankings using a maximum entropy model. In *Proceedings of the Stockholm Workshop on Variation within Optimality Theory*, ed. Jennifer Spenader, Anders Eriksson, and Östen Dahl, 111–120.

Grünwald, Peter. 1996. A minimum description length approach to grammar inference. In *Connectionist, statistical and symbolic approaches to learning for natural language processing*, ed. G. S. S. Wermter and E. Riloff, Springer Lecture Notes in Artificial Intelligence, 203–216. Springer.

Hale, Mark, and Charles Reiss. 1998. Formal and empirical arguments concerning phonological acquisition. *Linguistic Inquiry* 29:656–683.

Halle, Morris. 1962. Phonology in generative grammar. *Word* 18:54–72.

Hayes, Bruce, and Colin Wilson. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39:379–440.

Heinz, Jeffrey. 2007. The inductive learning of phonotactic patterns. Doctoral Dissertation, University of California, Los Angeles.

Heinz, Jeffrey, and William Idsardi. 2013. What complexity differences reveal about domains in language. *Topics in cognitive science* 5:111–131.

Heinz, Jeffrey, Gregory Kobele, and Jason Riggle. 2009. Evaluating the complexity of Optimality Theory. *Linguistic Inquiry* 40:277–288.

Heinz, Jeffrey, and Jason Riggle. 2011. Learnability. In *Blackwell companion to phonology*, ed. Marc van Oostendorp, Colin Ewen, Beth Hume, and Keren Rice. Wiley-Blackwell.

Horning, James. 1969. A study of grammatical inference. Doctoral Dissertation, Stanford.

Hsu, Anne S., and Nick Chater. 2010. The logical problem of language acquisition: A probabilistic perspective. *Cognitive Science* 34:972–1016.

Hsu, Anne S., Nick Chater, and Paul M.B. Vitányi. 2011. The probabilistic analysis of language acquisition: Theoretical, computational, and experimental analysis. *Cognition* 120:380 – 390.

Huffman, David A. 1952. A method for the construction of minimum-redundancy codes. *Proceedings of the I.R.E.* 1098–1102.

Inkelas, Sharon. 1995. The consequences of optimization for underspecification. In *Proceedings of NELS 25*, ed. Jill Beckman, 287–302. GLSA.

Jain, Sanjay, Daniel Osherson, James S. Royer, and Arun Sharma. 1999. *Systems that learn: An introduction to learning theory (learning, development and conceptual change)*. The MIT Press, 2nd edition.

Jarosz, Gaja. 2006a. Rich lexicons and restrictive grammars – Maximum Likelihood learning in Optimality Theory. Doctoral Dissertation, Johns Hopkins University, Baltimore, Maryland.

Jarosz, Gaja. 2006b. Richness of the base and probabilistic unsupervised learning in Optimality Theory. In *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology at HLT-NAACL 2006*, 50–59.

Jarosz, Gaja. 2010. Naive parameter learning for Optimality Theory – the hidden structure problem. Ms.

Katzir, Roni. 2014. A cognitively plausible model for grammar induction. *Journal of Language Modelling* 2:213–248.

Kiparsky, Paul. 2007. Panini's razor. Slides from a talk given at the First International Sanskrit Computational Linguistics Symposium, October 2007.

Kirkpatrick, Scott, C. Daniel Gelatt, and Mario P. Vecchi. 1983. Optimization by simulated annealing. *Science* 220:671–680.

Kolmogorov, Andrei Nikolaevic. 1965. Three approaches to the quantitative definition of information. *Problems of Information Transmission (Problemy Peredachi Informatsii)* 1:1–7. Republished as **?**.

Krämer, Martin. 2012. *Underlying representations*. Cambridge University Press.

Li, Ming, and Paul Vitányi. 2008. *An introduction to kolmogorov complexity and its applications*. Berlin: Springer Verlag, 3rd edition.

Manzini, M. Rita, and Kenneth Wexler. 1987. Parameters, binding theory, and learnability. *Linguistic Inquiry* 18:413–444.

de Marcken, Carl. 1996. Unsupervised language acquisition. Doctoral Dissertation, MIT, Cambridge, MA.

McCarthy, John J. 2005. Taking a free ride in morphophonemic learning. *Catalan Journal of Linguistics* 4:19–56.

Merchant, Nazarré Nathaniel. 2008. Discovering underlying forms: Contrast pairs and ranking. Doctoral Dissertation, Rutgers, The State University of New Jersey.

Moreton, Elliott. 2008. Analytic bias and phonological typology. *Phonology* 25:83–127.

Niyogi, Partha, and Robert C. Berwick. 1996. A language learning model for finite parameter spaces. *Cognition* 61:161–193.

Peña, Marcela, Luca Bonatti, Marina Nespor, and Jacques Mehler. 2002. Signal-driven computations in speech processing. *Science* 298:604–607.

Prince, Alan, and Paul Smolensky. 1993. Optimality theory: Constraint interaction in generative grammar. Technical report, Rutgers University, Center for Cognitive Science.

Prince, Alan, and Bruce Tesar. 2004. Learning phonotactic distributions. In *Constraints in phonological acquisition*, ed. René Kager, Joe Pater, and Wim Zonneveld, 245–291. Cambridge University Press.

Riggle, Jason. 2004. Generation, recognition, and learning in finite state Optimality Theory. Doctoral Dissertation, UCLA, Los Angeles, CA.

Riggle, Jason. 2006. Using entropy to learn OT grammars from surface forms alone. In *Proceedings of the 25th West Coast Conference on Formal Linguistics*, 346–353.

Rissanen, Jorma. 1978. Modeling by shortest data description. *Automatica* 14:465–471.

Rissanen, Jorma, and Eric Sven Ristad. 1994. Language acquisition in the MDL framework. In *Language computations: DIMACS Workshop on Human Language, March 20-22, 1992*, 149. Amer Mathematical Society.

Smith, Kirk H. 1966. Grammatical intrusions in the recall of structured letter pairs: Mediated transfer or position learning? *Journal of Experimental Psychology* 72:580–588.

Smolensky, Paul. 1996. The initial state and 'richness of the base' in Optimality Theory. Technical Report JHU-CogSci-96-4, Johns Hopkins University.

Solomonoff, Ray J. 1960. A preliminary report on a general theory of inductive inference. Technical Report ZTB-138, Zator Co., Cambridge, MA.

Solomonoff, Ray J. 1964. A formal theory of inductive inference, parts I and II. *Information and Control* 7:1–22, 224–254.

Steriade, Donca. 1995. Underspecification and markedness. In *The handbook of phonological theory*, ed. John Goldsmith, 114–174. Oxford: Blackwell.

Stolcke, Andreas. 1994. Bayesian learning of probabilistic language models. Doctoral Dissertation, University of California at Berkeley, Berkeley, California.

Tesar, Bruce. 2006. Faithful contrastive features in learning. *Cognitive Science* 30:863–903.

Tesar, Bruce. 2009. Learning phonological grammars for output-driven maps. In *Proceedings of NELS*, volume 39, 1013–0209.

Tesar, Bruce. 2014. *Output-driven phonology*. Cambridge University Press.

Tesar, Bruce, and Paul Smolensky. 1998. Learnability in Optimality Theory. *Linguistic Inquiry* 29:229–268.

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

Wallace, Christopher S., and David M. Boulton. 1968. An information measure for classification. *Computer Journal* 11:185–194.

Wexler, Kenneth, and Peter W. Culicover. 1980. *Formal principles of language acquisition*. Cambridge, MA: MIT Press.

Ezer Rasin
Department of Linguistics and Philosophy
Massachusetts Institute of Technology
Cambridge, MA 02139
USA
Email: rasin@mit.edu

Roni Katzir
Department of Linguistics and Sagol School of Neuroscience
Tel Aviv University
69978 Ramat Aviv
Israel
Email: rkatzir@post.tau.ac.il

Notes

[0] We thank Adam Albright, Dinah Baer-Henney, Iddo Berger, Bob Berwick, Noam Chomsky, Victoria Costa, Tova Friedman, Morris Halle, Jeffrey Heinz, Gaja Jarosz, Michael Kenstowicz, Paul Kiparsky, Andrew Nevins, Charlie O'Hara, Donca Steriade, the members of the Tel Aviv University Computational Linguistics Lab, and the audiences at Tel Aviv University, Hebrew University, and NecPhon 2013, as well as two anonymous reviewers for *LI*.

[1] Much of the literature on learning in OT assumes that the learner is provided with information about paradigmatically related forms, but here and in the remainder of this section we will base our discussion on observing raw input forms and will not assume access to additional information.

[2] This shift raises interesting issues regarding the architecture of the lexicon. We will revisit

this point briefly in section 3.3 below, but we hope that a more comprehensive discussion of the relevant issues and their resolution can wait for a separate occasion.

[3] Like the rest of the *ab*-nese example, the pattern of penultimate lengthening is highly artificial. Attested counterparts of this pattern include the interaction of stress and epenthesis in languages such as Mohawk and Yimas. See Alderete (1999) and Alderete and Tesar (2002) for discussion.

[4] To simplify the present discussion, we consider here only insertions and deletions as possible modifications.

[5] We ignore here the slight additional savings made possible by using a variable code length, for example through Huffman coding (Huffman, 1952). See Cover and Thomas (2006) for discussion.

[6] How many bits are implied by 'a few' and 'slightly' depends on the precise encoding scheme that is used. A concrete example (though for the learner rather than the phonologist) is provided in section 3 below.

[7] A reviewer suggests that the focus on economy (at the expense of restrictiveness) in early generative work may have come from an intuition that is close to the idea of Exception-based MDL (Li and Vitányi, 2008, pp. 397-8), where the grammar is stored alongside a list of exceptions.

[8] See Li and Vitányi (2008) for a detailed and thorough discussion of Kolmogorov complexity.

[9] The demand that the learner be able to accomplish its task given typical resource limitations (e.g., limited time, partial access to data, etc.) is sometimes formulated as a requirement of *feasibility*, as introduced by Chomsky (1965, p. 54) and discussed further by Wexler and Culicover (1980, pp. 17–22), among others.

[10] We will not attempt to speculate on the amount of data that the child may refer to (with one extreme being an unbounded batch learner, the other a memory-less online learner, and real life presumably somewhere in between). The learner presented here is a batch learner, but the amount of memory that it uses for the data in the following examples is relatively small (it is the size of the dataset, and the datasets are small). We hope that an investigation of the amount of data used by the human learner and of whether the current learner can be modified to match this memory constraint can wait for a separate occasion.

[11] The fixed alphabet could be part of the innate endowment of the learner. Alternatively, it could be learned during an earlier phase of learning. As far as the present discussion is concerned, all that matters is that the alphabet is fixed.

[12] In some of the simulations below we will slightly deviate from the encoding scheme presented here. When we do so, we will state the differences explicitly.

[13] Often that set contains one optimal output, but a tie between more than one candidate is possible in principle. We will assume that GEN allows for arbitrary insertions and deletions of segments.

[14] Since infinitely many candidates are generated by GEN, another possibility is that a tie is obtained between an infinite number of optimal candidates. This scenario can occur when epenthesis is not penalized by the grammar. If this happens, any output candidate o for a given UR u will have infinitely many variants that differ from o only in occurrences of epenthetic elements and thus incur the exact same violations of faithfulness constraints. If, for some most harmonic output candidate the markedness constraints fail to eliminate all but a finite number of epenthetic variants, the result will be a tie between the remaining infinitely many epenthetic variants. For our current purposes, we will assume that specifying a choice of one output from among an infinite set requires infinitely many bits of information. Another direction, not pursued here, is to decide on a non-uniform assignment of codes to candidates according to some enumeration. As far as we can tell, the choice does not affect the cases we discuss here. We hope that an exploration of this matter can wait for a separate occasion.

[15] The number of bits required to describe the data given the grammar is affected by the amount of data the learner is exposed to. By multiplying this factor by a large number we avoided working with large corpora that would have significantly increased the running time of our algorithm. We believe that the question of whether $|D{:}G|$ is indeed multiplied by a constant factor is an interesting question that should be empirically investigated. Currently, however, we have nothing substantial to say about this matter. Our results seem to be robust with respect the multiplication factor (small changes do not affect convergence).

[16] Given the initial temperature and the threshold, the number of iterations for our simulations

was a fixed 614,019. The simulations in sections 3.4 and 3.5, in which constraints were given in advance, converged even when the number of iterations was 92,099. We currently don't have a tight lower bound on the number of iterations required for convergence.

[17] In the literature following Smolensky (1996), an initial ordering of Markedness over Faithfulness ($M \gg F$) is often assumed as a means to confront the subset problem, but see Hale and Reiss (1998) for arguments in favor of a faithful initial state. See Albright and Hayes (2011) for further relevant discussion. On the current proposal, restrictiveness is obtained as a by-product of the MDL evaluation metric rather than as a property of the initial state.

[18] This is true, for example, if aspiration is represented as a separate segment, which is the somewhat simplistic representation we will use below. It is also true on various other, possibly more realistic ways to represent aspiration. It is possible, of course, to choose representations that make it cheaper to encode the presence of aspiration than its absence, but we find it hard to think of a justification for such a choice.

[19] We will not attempt to compare the segment-based encoding of the lexicon used in this simulation to the feature-based encoding that we use in all other simulations. We have successfully tested the segment-based encoding on all simulations presented in this paper (see Appendix 6), but chose to present the feature-based encoding as the default since it makes the connection to realistic phonological representations more transparent.

[20] The feature $[-spread\ glottis]$ constitutes the simplest choice for the learner to make here: it is the only way to refer to all underlying segments by using one feature.

[21] In Dell's original paper, only hypotheses corresponding to G_{simple} and G_{target} are considered. Dell proposes a learning strategy that always favors grammars that are more restrictive, and this strategy works well for cases in which these are the only choices. As we have seen, however, such a strategy will not work in a more general setting: it will have no reason to reject the problematic $G_{identity}$, which does not generalize at all, in favor of G_{target}.

[22] We presented each example 25 times to the learner in order to prevent over-generation by ensuring that optionality is not too cheap. Presenting each example once was not enough. Presenting

it 50 times lead to the same results as presenting it 25 times.

[23] This probabilistic version of OT is distinct from Stochastic OT (Boersma, 1998; Boersma and Hayes, 2001).

[24] This closely mirrors the minimization of $|D{:}G|$ alone under a description-length approach.

[25] What this approximation is varies. One option is to treat each element as being independently drawn from the lexicon according to a fixed probability distribution. An ML learner that makes such assumptions will memorize the empirical distribution of the elements in the data. The assumptions behind MLG are somewhat different: here the learner operates on the output of a morpheme analyzer, which leaves the ML learner the task of determining the conditional probabilities along the lines discussed earlier.

[26] If we are interested in learning the constraints themselves – as we were in our own learning examples – the inability of the ML learner to benefit from compressing the lexicon will be even more noticeable: a hypothesis with a listing of the surface forms (each with probability 1) and a single faithfulness constraint will always be optimal.

[27] Note that Riggle's version applies to lean lexicons, in which lexical entries are single URs, rather than the rich lexicons of Jarosz's system, in which lexical entries are distributions over possible URs.

[28] Allowing MAX and DEP to be ranked together is in line with certain variants of OT – see, in particular, Anttila (2007), who argues for the use of such rankings to account for optionality; a similar state of affairs is also possible within Stochastic OT (Boersma, 1998; Boersma and Hayes, 2001) – but we have chosen it here simply to make the presentation of the current point easier. We could have made the same point while adhering to strict linear orderings of the constraints, for example by considering a variant of *ab*-nese in which the following hold: two occurrences of *b* in a row are okay; three are not; an underlying *bbb* sequence can be repaired by a single insertion of *a* after the first occurrence of *b* but not after the second. A correct grammar would enforce the positional requirement on the insertion of *a*. For Riggle, however, the ranking **bbb* $\gg$ MAX $\gg$ DEP will do just as well, despite the fact that it overgenerates by allowing an underlying *bbb*

sequence to surface both (correctly) as *babb* and (incorrectly) as *bbab*.