LEARNABILITY UNDER OPTIMALITY THEORY

Iggy Roca, University of Essex

0. Introduction

It is nowadays taken as commonplace that the success of a linguistic theory cannot be measured exclusively in terms of formal elegance or descriptive coverage, but must be subject to the litmus test of learnability. The reason for this is obvious: language is a cognitive object, and it is the (self-imposed, but eminently reasonable) ultimate task of the linguist to describe this object, rather than its textual manifestations. It follows from this that the architecture that a linguistic theory projects onto language must be learnable under the usual conditions of inevitability, spontaneity, speed, perfection, and irrelevance of environmental conditions other than availability of ordinary language input to the child. The question to be asked in this paper is therefore whether or not the conception of grammar underlying Optimality Theory meets such learnability criteria.

This paper summarises the issues as discussed in McCarthy & Prince (1993), Prince & Smolensky (1993), and Tesar & Smolensky (1993), and is organised as follows. In section 2 I introduce Optimality Theory ('OT') for the benefit of the uninitiated reader, availing myself of a fragment of the syllabification grammar of Imdlawn Tashlhiyt ('IT') Berber, originally analysed in Dell & Elmedlaoui (1985, 1988) and previously presented in section 1 here, and I compare the OT descriptive achievements with those of the standard theory. In section 3, I explore the mechanism for the establishment of underlying forms under OT. Finally, in section 4 I scrutinise the learnability of language-specific constraint ranking, the keystone of OT grammars.

1. IT Berber syllables

The Imdlawn Tashlhiyt dialect of Berber, described in Dell & Elmedlaoui (1985, 1988), presents some remarkable syllabification characteristics. Consider first the data in (1) (syllable division is indicated by a dot throughout):

(1)	il.di	'he pulled'
	ir.ba	'he carried on his back'
	in.da	'he shook (milk)'
	im.da	'he was born out'
	iz.di	'he put together'
	iž.la	'he got lost'
	iɣ.za	'he digged'
	is.ti	'he selected'
	if.si	'he untied'
	ix.si	'he went out (fire)'

Such syllables as in (1) are as ordinary as any syllables can be.

In particular, all the words in question exhibit a syllabic structure VC.CV. CV is of course the universal core syllable, from which VC is readily obtainable by onset deletion and coda addition, as discussed in Clements & Keyser 1983. Note that although such (word-initial, or, more accurately, phrase-initial) onsetless syllables are optionally provided with a glottal stop onset (cf. Dell & Elmedlaoui 1985: 127, fn. 20), this is a late phonetic process that plays no role in syllabification as such, and consequently it will be ignored here.

Consider now the forms in (2):

```
(2) tr.glt     'you(sg) locked'
    ts.krt     'you(sg) did'
    tx.znt     'you(sg) stored'
    tz.dmt     'you(sg) gathered wood'
    tl.bžt     'you(sg) stepped onto'
    tr.kst     'you(sg) hid'
    tn.šft     'you(sg) grazed (skin)'
    tm.sxt     'you(sg) transformed'
```

These words are remarkable in that they have no vowels (NB. some ultra-short transitional vowels can reportedly be heard, but these are totally predictable from the phonetic context and have no phonological significance; cf. Dell & Elmedlaoui 1985: 116 ff.). What is striking is that, not only are these words pronounceable, indeed totally ordinary in IT Berber, but they are syllabified into two syllables each. This is of course a truly bizarre situation from the perspective of the speakers of 'regular' languages like English or German, but it is strongly substantiated by evidence from speaker intuition, emphasis spread, gemination, intonation, versification practice, and prosodic morphology (cf. Dell & Elmedlaoui 1985, 1988, Elmedlaoui 1985).

As a background to our analysis of IT Berber syllabification, we now provide two alternative formal representations of the (universal) basic syllable structure. The first, in (3a), is traditional, and purely stipulative, while the second, in (3b), is grounded on X-bar theory, and was originally proposed in Levin (1985):

```
(3) a. σ              b.   N''
       /\                  /\
      /  \                /  \
     O    R              /    N'
         /\             /    /\
        N  C           /    N  \

     p   e  n          p    e   n
```

On either representation, the core of each syllable is a vocalic nucleus, indeed the common situation universally. As is well-known, languages like English or German also allow sonorant consonants to constitute syllable nuclei under certain conditions (cf. e.g. *litt*[l], *butt*[n], *Neb*[l], *werf*[n] etc.). A close examination of the data in (2) will reveal that some of the proposed syllables contain a sonorant consonant, which can therefore reasonably be construed as the nucleus of the respective syllable. In other cases, such a nucleus is, even less plausibly, apparently constituted by a fricative. Still, this situation is also not totally unfamiliar to us, since fricative nuclei are found in English informal, relatively fast speech (cf. e.g. [s]*port* 'support' vs. [s]*port* 'sport'). The designated nuclei are made explicit in (4) by means of capitalisation:

(4) tR.gLt 'you(sg) locked'
tZ.dMt 'you(sg) gathered wood'
tL.bŽt 'you(sg) stepped onto'
tS.kRt 'you(sg) did'
tX.zNt 'you(sg) stored'
tR.kXt 'you(sg) hid'
tN.ŠFt 'you(sg) grazed (skin)'
tM.sXt 'you(sg) transformed'

From a traditional perspective, IT Berber could therefore be considered only to differ from such common-or-garden languages as English and French in its more liberal attitude to the nuclearisation of sonorant and fricative consonants. Consider, however, the forms in (5):

(5) mA.rA.tGt 'what will happen of you'
rA.tK.tI 'she will remember'
tF.tKt 'you suffered a sprain'

Here, alongside vocalic nuclei (mA, rA, tI), we find others consisting in an obstruent stop, whether voiced (tGt) or voiceless (tK, tKt). Clearly, we cannot go any further down on the scale of sonority, and consequently we must conclude that in IT Berber *all* segments qualify as syllable nuclei.

The apparently crazy situation just uncovered, where some (but obviously not all) of a word's consonants bear the syllable nucleus, may lead one to believe that this language simply has a few lexicalised consonantal nuclei, precisely as represented in (4) and (5) above. That this is not the case is forcefully brought out by the data in (6), which exhibit alternation with those in (1) above (non-nucleic first syllable consonant in (1) ~ nucleic first syllable consonant in (6)):

(6) tL.dI 'she pulled'
tR.bA 'she carried on her back'
tN.dA 'she shook (milk)'
tM.dA 'she was born out'
tZ.dI 'she put together'
tŽ.lA 'she got lost'
tY.zA 'she digged'
tS.tI 'she selected'
tF.sI 'she untied'
tX.sI 'she went out (fire)'

Indeed, were it to be just a matter of arbitrary lexicalisation, we would expect the distribution of such syllabic consonants to be random. However, as will now be shown, IT syllabification is constrained by the two principles in (7):

(7) IT syllabification principles:

i. no hiatus (i.e. *NN)

ii. maximisation of number of syllables

The principle of no-hiatus entails obligatoriness of onsets in all positions but phrase-initially. As a consequence, the parsings in (8a), but not those in (8b), will be legitimate (NB. /w/ = non-nucleic /u/; /y/ = non-nucleic /i/):

(8)	a. tI.wN.tAs	'you climbed on him'	b.	*tI.Un.tAs
	rA.yMm.$_{y}$I	'he will grow'		*rA.Imm.$_{y}$I

In turn, the principle of syllable maximisation implies both syllabification recursiveness and minimisation of coda construction (subject to some further constraints on the structure of the syllable). In a nutshell, the idealised effect of the IT Berber syllabification algorithm is as schematised in (9):

(9) IT core syllable: CV (subsequently incremented)

preferred syllabification: CV.CV.CV.CV.CV....
ON.ON.ON.ON.ON....

Underpinning the succession of core syllables CV.CV. ... is of course the sonority substance of segments and the universal principle of sonority dispersion (cf. Clements 1990), according to which sonority differences between onset and nucleus tend to be maximised, while sonority differences between nucleus and coda tend to be kept to a minimum (∅ is obviously the optimal such minimum). The role of sonority in IT Berber syllabification goes however beyond such universal effects, as illustrated in (10) (NB. onsetless stops are eventually desyllabified and incorporated into the adjacent syllables, as commented on in Dell & Elmedlaoui 1985: 127, fn. 16; we abstract away this eventuality for the sake of clarity and simplicity of exposition; note interestingly that the pre-desyllabification level is made use of in Berber poetry, as discussed in Dell & Elmedlaoui 1988):

(10)	a. T.zMt	'she is stifling'	b.	*tZ.mT
	rAt.lU.lT	'you will be born'		*rA.tL.wLt

Here, the parsings in a. and b. both comply with the basic syllabification requirements of the language (remember that onsets are not obligatory phrase-initially, since no hiatus results). This notwithstanding, only the parsings in a. are legitimate, for reasons relating to sonority, as will become clear directly.

As a preliminary, I shall make explicit the (unremarkable) IT Berber sonority hierarchy (I ignore a few additional, exotic IT Berber consonants for the sake of graphic simplicity):

(11) IT Berber sonority ranking:

low vowel (a) >

high vowels (i, u) >

liquids (l, r) >

nasals (m, n) >

voiced fricatives (z, ž, γ) >

voiceless fricatives (f, s, š, x) >

voiced stops (b, d, g) >

voiceless stops (t, k, q)

Armed with such sonority-based grading of IT Berber segments, I will now provide the basic syllabification algorithm of the language, nucleus assignment being indeed algorithm-governed, and not lexical, as hinted at above:

(12) core syllabification algorithm (Dell & Elmedlaoui 1985, 111):

> 'associate a core syllable with any sequence (Y)Z, where Y can be any segment and Z is a segment of type T, where T is a variable to be replaced by a certain set of feature specifications [= descending sonority, IMR]'

The segment on the lefthand side, Y, can *only* be missing phrase-initially, as we already know, an important restriction which is however not made explicit by Dell & Elmedlaoui in the algorithm itself. The following derivations illustrate the workings of this algorithm (the forms in parentheses result from the operation of coda formation, not part of the core algorithm; nucleus parsing is symbolised by means of a vertical tree line):

```
(13) trba              tzmt               ratlult

       /|               /|                /|
     trbA              tzMt               rAtlult      step 1


     /|/|              |/|                /| /|
     tRbA              TzMt               rAtlUlt      step 2

                                          /| /|/|
                                          rAtlUlT      step 3

                       |/|\               /|\/|/|
                       (TzMt)             (rAtlUlT)    coda

     'she carries      'you will be born' 'she is stifling'
     on her back'
```

This algorithm is still ambiguous when applied to such underlying forms as /rksx/ 'I hid' and /bainn/ 'they (m.) appear', as illustrated in (14):

(14) a. R.kSx 'I hid' b. *Rk.sX
bA.yNn *'they (m.) appear'* **bAy.nN*

The difference between the two sets of outputs is a function of directionality (L-to-R in a. vs. R-to-L in b.), which must accordingly also be specified in the procedure (L-to-R). This enrichment is however still insufficient, and further conditions need to be imposed. For instance, we must rule out ambisyllabicity (*(X (X) X)), and prevent destruction of structure ((N)XX -/-> (ONC)), in line with the so-called Free Element Condition (cf. Prince 1985). Finally, while the high vocoids /i/, /u/ can be parsed in the onset when required by the algorithm, as we have seen, the low vocoid /a/ is only parsable in the nucleus, as illustrated by the following data from the similar Ait Seghrouchen dialect, analysed in Guerssel (1985):

(15) inna-ax 'he told us' -> in.na.**y**ax
a aryaz 'man (voc.)' -> a.**y**ar.yaz

As can be seen, hiatus is resolved by consonant epenthesis ([y], highlighted for convenience), rather than by the assignment of /a/ to the onset (cf. e.g. *In.nA.aX).

We shall now take brief stock and bring together the core syllabification machinery of IT Berber under the standard analysis of Dell & Elmedlaoui (1985):

(16) standard syllabification machinery of IT Berber:

i. algorithm (12)

ii. exhaustivity condition

iii. free element condition

iv. ambisyllabicity ban

v. /a/ -> Nuc (i.e. *.aN.)

The problem with this procedure concerns not so much its obvious complexity (only partially alleviated by the universality of some of its components), but specifically the apparent lack of connection between the various rules and conditions, the relation between which, if existent, is anything but obvious. As we will see in the next section, this difficulty is elegantly circumvented by OT.

2. Optimality Theory

The principles of OT are lucidly expounded in McCarthy & Prince (1993) and Prince & Smolensky (1993), and will now be summarised for the reader's convenience.

The basic mechanism of OT is extremely simple, and is made up of two components, viz. Gen and a set of constraints. I will comment on these in turn.

Gen (short for 'generator') is a device parsing each of a set of universal inputs into a (universally acceptable) set of outputs. Thus, for instance, given a string of segments, Gen will produce a sequence of universally (NB. not necessarily language-specifically) well-formed syllables (the question of which syllables are universally well-formed is itself of course still open, at least on the edges; the full answer to this question, whatever it may be, will thus simply be incorporated into the body of Gen, according to OT tenets). An important corollary of the restriction of Gen activity to parsing is that OT does not countenance physical deletion as such, and so any input will be contained in each of its outputs (the 'Principle of Containment').

If the grammar of all languages consisted exclusively of Gen, all languages would be identical. More precisely, there would literally be only one language, subject to random variation, given the relative unrestrictedness of Gen. Clearly, therefore, further principles are necessary to reflect both (relative) language-internal invariance and cross-linguistic variation. The key feature of the OT framework is that all such principles are couched in terms of (positive or negative) constraints (NB. not rules), which are moreover postulated to be universal, hence not learned (Gen is obviously also part of Universal Grammar). Again, given the universality of the constraints, the prediction is that all languages will be identical (equivalently, only one language will be in existence). This prediction is of course counterfactual, and OT consequently allows for the *ranking* of constraints according to language-specific stipulation, the universal, inviolable principle being that compliance with higher ranked constraints takes precedence over compliance with their lower ranked counterparts. In this way, the preferred output will be the *optimal* output (i.e. the output that best meets the principle of higher ranking priority), rather than the *perfect* output (i.e. the output that violates no constraints), which is more often than not simply unobtainable (note also, and importantly, that, because all constraints apply to the output of Gen simultaneously, OT countenances no derivations).

I shall now illustrate the workings of this simple model by applying it to our familiar IT Berber data. Let us first formalise a couple of constraints playing a basic role in syllabification in general (for discussion of these constraints and other parts of the theory, and for a more extended OT analysis of IT Berber, see Prince & Smolensky 1993):

(17) two (output) constraints:

a. Onset = syllables must have onsets

b. Nuclear Harmony (NucHarm) =

if |x| > |y| then Nuc/x >- Nuc/y

The interpretation of these constraints is straightforward. In particular, Onset simply requires syllables to have onsets (remember that the most unmarked core syllable is CV, not V), in a manner equivalent to the Minimal Onset Satisfaction principle of Roca (1994) (in turn incorporating insights from Selkirk 1982, Steriade 1982, etc.). NucHarm dictates that, given two segments x and y, such that x is more sonorous than y ('|x| > |y|'), then x is a better (or 'more harmonic') nucleus than y ('Nuc/x >- Nuc/y'). Again, this is clearly a simple rephrasing of a universal principle of markedness.

Let us next look at the interaction between these two constraints in IT Berber. We shall postulate the ranking in (18):

(18) Onset >> NucHarm

In prose, satisfaction of Onset must take priority ('>>') over satisfaction of NucHarm.

In order to justify this ranking, we shall examine a tableau (= a table displaying a set of possible candidate parses output by Gen, and their respective fate under each constraint) for the underlying form /txznt/ 'you stored' (each * represents one constraint violation; an exclamation mark ! signals that the corresponding violation mark is fatal, i.e. that it effectively disposes of the candidate being evaluated; the optimal candidate is marked with an arrow head '>'):

(19)

<table>
<tr><th>candidates</th><th colspan="2">constraints</th><th>comments</th></tr>
<tr><th></th><th>Onset</th><th>NucHarm</th><th></th></tr>
<tr><td>T.X.Z.N.T</td><td>*!***</td><td>nzxtt</td><td>NucHarm irrelevant</td></tr>
<tr><td>T.xZ.Nt</td><td>*!</td><td>n z t</td><td>NucHarm irrelevant</td></tr>
<tr><td>> tX.zNt</td><td></td><td>n x</td><td>optimal</td></tr>
<tr><td>Tx.zNt</td><td></td><td>n t!</td><td>|n| = |n|, |t| < |x|</td></tr>
<tr><td>tXz.nT</td><td></td><td>x! t</td><td>|x| < |n|, t irrelevant</td></tr>
</table>

Clearly, the two constraints in (18) will not be sufficient to account for all and only the existing types of syllables, both universally and in IT Berber. Accordingly, further syllabification-related constraints must be postulated, as follows:

(20) further syllabification constraints:

a. Parse =
segmental material must be incorporated into syllabic structure

b. Fill =
syllabic structure can only be built on segmental material

c. ~Coda =
there is no coda

d. ~M/a =
/a/ is not parsed in the syllable margin

Parse and Fill are the two 'faithfulness' constraints enforcing isomorphy between underlying and surface representations. ~Coda (or No-Coda) again aims at the universally unmarked syllable CV. Finally, ~M/a is an extreme instantiation of NucHarm, simply excluding the maximally sonorous segment /a/ from the syllable margins (thus forcing its parsing in the syllable nucleus).

I now illustrate the ranking of these constraints, and its consequences, in IT Berber. For convenience, I shall use the abstract underlying sequences /naa/, /nia/, /nai/, and /tk/ (dotted vertical lines between constraints in tableaux conventionally represent equality of ranking; a continuous line indicates a left-to-right hierarchical relation):

(21)

	ONS	PARSE	FILL^{nuc}	~M/a	FILL^{onset}	NucHarm	~CODA
> nA.[]A					*	a a	
naA				*!		a	
nA<a>		*!				a	
nA.A	*!					a a	
> N.iA						a n	
nI.[]A					*!	a i	
nIa				*!		i	*
> nAi						a	*
nA.[]I					*!	a i	
naI				*!		i	
> tK						k	
t[]k			*!				*

As can be seen, all the facts of IT Berber syllabification are accounted for satisfactorily. The obvious advantage of this analysis over its counterpart in the standard theory lies in the simplicity and homogeneity of its machinery. In particular, the constraint inventory is universal, all the desired facts then simply falling out of a given language-specific ranking, as has been shown (note that the Onset >> NucHarm ranking needs reversing in phrase-initial position; this undesirable twist is however replicated in the unsightly standard condition '(Y) = Ø only phrase-initially'). By contrast, the standard machinery displayed in (16) above is disturbingly diverse, as we pointed out at the time.

3. Underlying Representations in OT

As mentioned in the introduction, the acid test for theory evaluation is not so much descriptive success (which OT manifestly achieves for the data under scrutiny), but psychological plausibility from the perspective of ordinary language learning. This issue is directly addressed in Tesar & Smolensky (1993), who taxonomise knowledge of language under OT as in (22):

(22) knowledge of language under OT

i. Gen (= mapping of universal inputs onto universal outputs)
ii. constraints (on Gen outputs)

iii. underlying forms
iv. constraint ranking

On these, i. and ii. are part of Universal Grammar, and thus unlearned. The learning problem therefore only concerns iii. and iv., to which we now turn.

The issue of how underlying forms are set up by the learner, typically overlooked in the phonological literature, is specifically addressed in Prince and Smolensky (1993) under the label 'lexicon optimisation'. In particular, these authors propose to constrain underlying forms by means of the following principle:

(23) Lexicon Optimisation (Prince and Smolensky 1993:192):

> 'Suppose that several different inputs I_1, I_2, ..., I_n when parsed by a grammar G lead to corresponding outputs O_1, O_2, ..., O_n, all of which are realised as the same phonetic form ϕ - these inputs are all *phonetically equivalent* with respect to G. Now one of these outputs must be the most harmonic, by virtue of incurring the least significant violation marks: suppose this optimal one is labelled O_k. Then the learner should choose, as the underlying form for ϕ, the input I_k'.

This principle can effectively be interpreted as implementing respect for PARSE and FILL, the faithfulness constraints (so-called precisely because their role is indeed to ensure a faithful reproduction of the underlying representation in the surface form), by the learner setting up underlying representations, as we shall now see.

Take for example the sequence [CV], which constitutes the universally preferred syllable. Other things being equal, it of course makes sense to postulate /CV/ as the corresponding underlying representation, as any self-respecting phonologist undoubtedly knows. But how is this underlying form arrived at by the learner from the perspective of OT? Remember that the relevant constraints are PARSE and FILL, the faithfulness constraints. I illustrate various logical possibilities in (24):

(24) possible sources of [CV]:

surface parse	hypothesised URs	constraint violations
CV	/CV/	
CV.<V>	/CVV/	*PARSE
<C>.CV.<V>	/CCVV/	**PARSE
<C><C>.CV.<V><V>	/CCCVVV/	****PARSE
C[]	/C/	*FILL
[]V	/V/	*FILL
[] []	/0/	**FILL
[]V<C>	/VC/	*FILL, *PARSE
<V>C[]	/VC/	*PARSE, *FILL
etc.		

As can be seen, all but the first of these UR candidates, /CV/, incur violations of the two given constraints. Consequently, by the Lexicon Optimisation Principle in (23), /CV/ will be selected as the underlying form of [CV], all according to common sense and phonologist's intuition, as pointed out.

The irrelevance of the remaining constraints to lexical optimisation, in the sense of (23) above, will now be demonstrated. Suppose that the input datum is [VC]:

(25) possible sources of [VC]:

surface parse	hypothesised URs	constraint violations
VC	/VC/	*ONSET, *~CODA
<C>V[]	/CV/	*ONSET, *~CODA, *PARSE, *FILL
<C>VC	/CVC/	*ONSET, *~CODA, *PARSE
[]C	/C/	*ONSET, *~CODA, *FILL
V[]	/V/	*ONSET, *~CODA, *FILL
<CC>V<V>C<C>	/CCVVCC/	*ONSET, *~CODA, ****PARSE
etc.		

As can be seen, any deviation of the UR from the surface form automatically results in the violation of the faithfulness constraints. Additional constraints are contravened by parsings yielding outputs at variance with universal unmarkedness, but this situation cannot be repaired by tinkering with the UR, since constraint violation is obviously computed on the surface form, and this is given. The upshot of the discussion is therefore that the Lexicon Optimisation Principle (23) will force the selection of URs identical to the surface realisation, a result no doubt highly encouraging for the hard-nosed empiricist phonologist all along suspicious of SPE-type abstractness.

The joy of such a phonologist will however be short-lived, since things become considerably more complex (and more lively) as soon as alternation (a fact of life for natural languages) is brought into the picture. In particular, Prince & Smolensky formulate a

Minimal Redundancy Principle disfavouring the presence of lexical material:

(26) Minimal Redundancy Principle (Prince & Smolensky 1993: 195):

> To the maximal extent possible, information should be excluded from the lexicon which is predictable from grammatical constraints

A more general (and more extreme), optimality-couched version of this principle is given in (27) in the form of a negative constraint:

(27) ~Spec (Prince & Smolensky 1993: 196):

> Underlying material must be absent

The relevance of these additional considerations will now be exemplified with the passive conjugation allomorphy of Maori, as described in the by now classic account of Hale (1973). The basic morphology is presented in (28):

(28)	UR		surface		
	stem	affix	inflected	uninflected	
	CVCV	+V	CV.CV.V	CV.CV	
	kite	+a	ki.te.a	ki.te	'to embrace'
	patu	+a	pa.tu.a	pa.tu	'to kill'

As can be seen, inflection consists of a suffix *-a*, added to the (CVCV) root. The URs postulated simply follow from the principle of Lexical Optimisation in (23), specifically from the action of the faithfulness constraints.

Consider now the forms in (29):

(29)	UR		surface		
	stem	affix	inflected	uninflected	
	CVCV	+CVV	CV.CV.CV.V	CV.CV	
	wero	+hia	we.ro.hi.a	we.ro	'to stab'
	tohu	+ŋia	to.hu.ŋi.a	to.hu	'to point out'
	hopu	+kia	ho.pu.ki.a	ho.pu	'to catch'

The postulated URs are again faithful to the surface forms. The difference with (28) lies in the suffix, which is now CVV, with the additional complication that its initial consonant is seemingly unpredictable. Strict adherence to the faithfulness constraints will thus inevitably lead to the establishment of a sizeable number of conjugation classes, with the corresponding multiplication of underlying suffixes (i.e. one for each consonant), against the grain of the economy principles (26) and (27).

An alternative analysis circumventing both these difficulties is however available:

(30)

	UR		surface		
	stem	affix	inflected	uninflected	
	CVCVC	+VV	CV.CV.CV.V	CV.CV.<C>	
	weroh	+ia	we.ro.hi.a	we.ro	'to stab'
	tohuŋ	+ia	to.hu.ŋi.a	to.hu	'to point out'
	hopuk	+ia	ho.pu.ki.a	ho.pu	'to catch'

What we are now doing is assigning the ostensibly suffix-initial consonant to the stem. The immediate consequence of this move is the reduction of the suffixal allomorphy to /a/ and /ia/. This remaining allomorphy is moreover reducible to rule (/a/ after a vowel, and /ia/ after a consonant), and consequently, we can do away with all conjugation classes.

We are still seemingly paying the price of a deletion rule disposing of the underlying stem-final consonant in uninflected forms. Prince & Smolensly, however, point out that such deletion will fall out of the (inviolable) syllabic template of Maori: (C)V. In particular, because codas are disallowed across the board in this language, the constraint ~CODA will be undominated, i.e. placed at the top in the ranking (this ranking is of course still unavailable at the time URs are being learnt: Prince & Smolensky are simply anticipating this result at this point; note however that the question still remains as to what makes the learner decide precisely for the desired UR in the absence of the relevant ranking information). Such a position in the ranking (crucially shared with FILL) will ensure deletion (more precisely, underparsing) of the underlying stem-final consonant of verbs word-finally, and thus no specific rule or equivalent will be necessary to achieve this result:

(31)

a.

	FILL	~COD	PARSE	ONSET
> we.ro.hi.a				*
we.ro.h<i>a			*!	
we.ro.<h>i.a			*!	**
we.roh.i.a		*!		**

b.

	FILL	~COD	PARSE	ONSET
we.ro.h[]	*!			
> we.ro.<h>			*	
we.roh		*!		

Underlying forms such as /weroh/ are therefore optimal in the context of the observed alternation, even though they infringe the Lexicon Optimisation Principle (23). In particular, this solution is superior to the one that multiplies the UR of suffixes, examined above, as a consequence of the Minimal Redundany Principle (26) (or its bare bone constraint incarnation in (27). This means that, in the event of conflict between these two principles, the Minimal Redundany Principle (26) emerges victorious, since it is precisely this principle that licenses violation of the faithfulness conditions, and thus the existence of URs diverging from surface forms.

The implications of this scenario for learnability are obvious. We must assume that both the Minimal Redundancy Principle (26) and the Lexicon Optimisation Principle (23) are utilised by the learner as part of the general language learning algorithm, and that this algorithm awards greater weighting to the former constraint (itself crucially restrained by the caveat 'to the maximal extent possible', which obviously stands in the way of wild suppression of surface substance underlyingly). We have shown that, given this assumption, the desired results follow automatically from the set of available data. Note, however, that in the real world such data clearly do not become accessible instantaneously, and therefore the acquisition of URs will necessitate gradual exposure to a rich array of data over time.

4. Learnability of Constraint Ranking

We now turn our attention to the issue of learnability of constraint ranking, specifically addressed in Tesar & Smolensky (1993).

As will be recalled, Tesar & Smolensky (1993) assume that the basic material available to the learner are surface forms (given) and their corresponding URs (arrived at in the way described in the previous section). Thus, assuming the surface form [tola] for some hypothetical language L_1, the specific evidence directly available to the learner will be as in (32) ('[]' symbolises the abstract segment resulting from overparsing; we are obviously assuming that epenthetic consonants are realised as [t] in this language):

(32) positive evidence:

```
/VCVC/ -> []V.CV.<C>          L1
 olas     []o.la.<s>

          [to.la]
```

The proposed underlying form /olas/ is of course not faithful to the surface form [tola], and therefore we must assume the existence of alternations in the general data body of L_1 motivating such a degree of abstractness, in line with our discussion in the previous section (remember, in particular, that infringements of Lexicon Optimisation must be offset by successes of Minimal Redundancy).

The parsing corresponding to L_1 in (32) is thus empirically legitimised. By contrast, all other parsings generated by Gen from the given UR will be in conflict with the facts. Such parsings (corresponding to languages L_2, L_3, L_4, etc., ad infinitum given FILL, all distinct from L_1), displayed in (33) below, constitute therefore negative evidence readily inferable by the learner (NB. vowel epenthesis is assumed to be implemented as [i] in these languages):

(33) negative (inferred) evidence:

```
 V.CVC                          L2
 o.las

*[o.las]


 <V>.CV<C>                      L3
 <o>.la <s>

*[la]


 <V>.CV.C[]                     L4
 <o>.la.s[]

*[la.si]

                                 :
ad infinitum                     :
```

As can be seen, the legitimate and illegitimate forms (all of them parse 'candidates') in (32) and (33), respectively, incur (or may incur) a number of constraint violations. The list of constraints being universal, and therefore unlearned, it will be possible for the learner readily to verify such violations, as we now represent in the table (NB. not tableau!) in (34):

(34) L_1 candidate evaluation:

	ONS	~COD	FILL^{nuc}	PARSE	FILL^{ons}
a. *V.CVC	*	*			
b. *<V>.CV.<C>				**	
c. *<V>.CV.C[]			*	*	
d. []V.CV.<C>				*	*

The table simply displays which of the various constraints are violated by each candidate. Such candidate evaluation is of course completely independent of constraint ranking (the constraints are obviously still unranked, the whole point of the exercise being precisely that of arriving at a ranking on the basis of the raw data).

The data in (32) and (33) above (all accessible to the learner, as we have seen) can readily be arranged as data pairs, as in (35):

(35) data pairs:

V.CVC -< []V.CV.<C>

<V>.CV.<C> -< []V.CV.<C>

<V>.CV.C[] -< []V.CV.<C>

In particular, each possible but empirically unsubstantiated parsing of the UR is stated as less harmonic ('-<') than the attested parsing (corresponding to L_1 in our example), as corresponds to the general scheme in (36) (in Tesar & Smolensky's terminology, 'suboptimal' refers to the parsings that yield illegitimate forms, and 'optimal' to the parsing corresponding to the attested form):

(36) subopt_i -< opt

Specifically, each suboptimal parsing ('subopt_i') is less harmonic than its optimal ('opt') congener (indeed by definition!).

Now, such data pairs, automatically derived from the conjunction of the positive and negative evidence, as we have seen, contain the seed of constraint ranking. In particular, given the logic of

OT, where harmony is a function of constraint ranking, the suboptimal candidates can only be so if the constraints they violate (as manifested in the marks they incur) are ranked higher than the constraints violated by the optimal candidate:

(37) marks(subopt) >> marks(opt)

The next step in the procedure consequently involves the comparison of the constraint violations of each of the suboptimal candidates with those incurred by the optimal candidate, as illustrated in (38) (the labels a ... d refer to the lines in table (34) above):

(38) L_1 mark-data pairs:

	$subopt_i$ -< opt_i	marks(subopt)	marks(opt)
a -< d	V.CVC -< []V.CV.<C>	{*ONS, *~COD}	{*PARSE, *$FILL^{ons}$}
b -< d	<V>.CV.<C> -< []V.CV.<C>	{*PARSE, *PARSE}	{*PARSE, *$FILL^{ons}$}
c -< d	<V>.CV.C[] -< []V.CV.<C>	{*PARSE, *$FILL^{nuc}$}	{*PARSE, *$FILL^{ons}$}

As can be seen in (38), it is possible for the same constraint to be violated by both the optimal and the suboptimal candidate in the same line. Such a situation comes under the remit of the Cancellation/Domination Lemma of Prince & Smolensky (1993):

(39) Cancellation/Domination Lemma (Prince & Smolensky 1993):

> Suppose two parses B and C do not incur identical sets of marks. Then B >- C if and only if every mark incurred by B which is not cancelled by a mark of C is dominated by an uncancelled mark of C

In particular, harmony relations are, reasonably, only established on the basis of uncancelled marks. Equivalently, marks incurred on the same constraint by both candidates in the same line have no effect on their relative harmony. Accordingly in the next step in the procedure, marks common to both candidates in each line are cancelled from the table of mark-data pairs (cancelled marks have been struck out, for greater visual clarity):

(40) common mark cancellation:

	subopt$_i$ -< opt$_i$	marks(subopt)	marks(opt)
a -< d	V.CVC -< []V.CV.<C>	{*ONS, *~COD}	{*PARSE, *FILL^{ons}}
b -< d	<V>.CV.<C> -< []V.CV.<C>	{*~~PARSE~~, *PARSE}	{*~~PARSE~~, *FILL^{ons}}
c -< d	<V>.CV.C[] -< []V.CV.<C>	{*~~PARSE~~, *FILL^{nuc}}	{*~~PARSE~~, *FILL^{ons}}

The table in (40), processed directly from the raw evidence, as we have seen, contains all the data on which the constraint ranking learning algorithm will operate.

In the initial state of this algorithm, all constraints are equally ranked (in fact, they are supposedly unranked), as corresponds to their neutral state in Universal Grammar (notice that this points to a Superset Principle, by which languages hypothesised at earlier learning stages are supersets of languages hypothesised later):

(41) constraint ranking learning algorithm ('H' = 'hierarchy'):

initialisation: H = H$_o$
{ONS, ~COD, PARSE, FILL^{nuc}, FILL^{ons}}

In the next step, the first line in table (40) (a -< d) is examined. FILL^{ons} and PARSE are violated by the optimal candidate. In the logic of OT, this means that they must be out-ranked by the constraints violated by the suboptimal candidate, viz. ONSET and ~CODA. In the model of Tesar & Smolensky, this situation induces rearranging of the present constraint ranking. In particular, FILL^{ons} and PARSE are demoted to the next lower rung of the hierarchy:

(42) constraint demotion (for a -< d):

a:	{ONS,~COD, FILL^{nuc}}	highest-ranked constraints
	{FILL^{ons}, PARSE}	not-yet-ranked constraints

Any suboptimal candidate incurring violation of one of the constraints ranked highest in (42) is automatically accounted for by the current ranking, for the simple reason that such a constraint is already ranked higher than FILL^{ons} and PARSE, which have been demoted in (42). Consequently, any line containing a violation of any of the highest ranking constraints can be eliminated from the computation. This obviously disposes of line a. in table (40); also of line c., where the suboptimal candidate violates highest-ranked $FILL^{nuc}$. The mark table is therefore reduced as in (43):

(43) reduced mark table:

	$subopt_i$ -< opt_i	$marks_{subopt}$	$marks_{opt}$
b -< d	<V>.CV.<C> -< []V.CV.<C>	{~~*PARSE~~, *PARSE}	{~~*PARSE~~, *$FILL^{ons}$}

Notice that PARSE, violated by the suboptimal candidate in the remaining line, is not included in the set of highest ranked constraints in (42).

As expected, the familiar demotion procedure is reapplied to the reduced mark table in (43):

(44) constraint demotion (for b -< d):

b: {PARSE} — next-highest ranked constraints

{$FILL^{ons}$} — not-yet-ranked constraints

The ranking of PARSE above $FILL^{ons}$ induces removal of line b in (43). After the removal of this line, the mark table becomes empty, in the obvious way:

(45) reduced mark table:

$subopt_i$ -< opt_i	$marks_{subopt}$	$marks_{opt}$

Further application of constraint demotion to this table simply terminates the algorithm, since there are no constraints left to be demoted:

(46) constraint demotion:

{$FILL^{ons}$} — next-highest ranked constraints

{} — not-yet-ranked constraints

The resulting stratified hierarchy is thus as in (47):

(47) L_1 stratified hierarchy:

$\{ONS, \sim COD, FILL^{nuc}\} >> \{PARSE\} >> \{FILL^{ons}\}$

The crucial points are that the algorithm terminates, and that the outcome has been arrived at deterministically. The constraint ranking responsible for any particular form is therefore logically learnable.

A different consideration concerns learnability load. In particular a logically learnable ranking may not be learnable in real time, specifically in the real acquisition time pertinent to the acquisition of language.

Tesar & Smolensky (1993) contend however that this is not the case. The steps in their argument are as follows. First, each pass through the table of mark-data pairs must output at least one constraint. If so, the number of passes cannot be greater than the number of (universal) constraints ('N_{constr}'). Second, the number of steps in each pass cannot be greater than the number of uncancelled marks in the table, i.e. maximally N_{constr} x N_{pairs} ('N_{pairs}' = number of lines in the mark-data table). Consequently, the total number of steps involved in the implementation of the algorithm is as in (48):

(48) learnability load: $(N_{constr})^2\ N_{pairs}$

The product of this equation is likely not to exceed a few thousand, a figure readily copable with by the neurological machinery of man. Consequently, not only is constraint ranking logically learnable, but it also appears to be learnable under the real-world conditions relevant to language. As things stand at the moment, however, this conclusion applies to lexical items piecemeal, and it remains to be demonstrated that the induction of the general constraint ranking relevant to any one grammar indeed is a feasible task. Crucially, learning of a language involves learning of its lexical items, in the sense that it cannot be said that the language is known unless the lexical items are known. Clearly, if the lexical items are known, the overall ranking, deterministically derived from such items, can be learnt.

Acknowledgements

Previous incarnations of this paper were presented at the University of Essex, the University of Stuttgart, and FAS, Berlin. I am grateful to the respective audiences for questions and comments, and to Greg Dogil and Richard Wiese for kindly inviting me to talk. I also thank Sylvia Löhken for giving me no choice about writing out the paper, and Martin Atkinson for comments on a prior version that considerably tempered my earlier sanguinity. All errors are mine.

References

Clements, G.N. (1990) 'The Role of the Sonority Cycle in Core Syllabification', in J. Kingston and M. Beckman, *Papers in Laboratory Phonology I*, Cambridge: Cambridge University Press, 283-333.

Clements, G.N. & S.J. Keyser (1983) *CV-Phonology*, Cambridge, MA: MIT Press

Dell, F. & M. Elmedlaoui (1985) 'Syllabic Consonants and Syllabification in Imdlawn Tashlhiyt Berber', *Journal of African Languages and Linguistics* 7: 105-130.

Dell, F. & M. Elmedlaoui (1988) 'Syllabic Consonants in Berber: Some New Evidence', *Journal of African Languages and Linguistics* 10: 1-17.

Elmedlaoui, M. (1985) Le parler berbère chleu d'Imdlawn: segments et syllabation, Doctorat de Troisième Cycle, Université de Paris VIII, Saint-Denis.

Guerssel, M. (1985) 'Glides in Berber and Syllabicity', *Linguistic Inquiry* 17: 1-12.

Levin, J. (1985) A Metrical Theory of Syllabicity, PhD dissertation, MIT.

McCarthy, J. & A. Prince (1993) Prosodic Morphology I, ms., University of Massachusetts at Amherst and Rutgers University.

Prince, A. (1985) 'Improving Tree Theory', in *Proceedings of the XIth Annual Meeting of the Berkeley Linguistics Society*, 471-90.

Prince, A. & P. Smolensky (1993) Optimality Theory, ms., Rutgers University and University of Colorado at Boulder.

Roca, I. (1994) *Generative Phonology*, London: Routledge.

Selkirk, E. (1982) 'The Syllable', in H. van der Hulst and N. Smith (eds.), *The Structure of Phonological Representations (Part II)*, Dordrecht: Foris, 337-83.

Steriade, D. (1982) Greek Prosodies and the Nature of Syllabification, PhD dissertation, MIT.

Tesar, B. & P. Smolensky (1993) 'The Learnability of Optimality Theory', ms., University of Colorado at Boulder.