2. Optimality in Grammar: Core Syllabification in Imdlawn Tashlhiyt Berber

Here we argue that certain grammatical processes can only be properly understood as selecting the *optimal output* from among a set of possibilities, where the term *optimal* is defined in terms of the constraints bearing on the grammatical domain at issue.

2.1 The Heart of Dell & Elmedlaoui

The Imdlawn Tashlhiyt dialect of Berber (ITB) has been the object of a series of remarkable studies by François Dell and Mohamed Elmedlaoui (Dell & Elmedlaoui 1985, 1988, 1989). Perhaps their most surprising empirical finding is that in this language any segment — consonant or vowel, obstruent or sonorant — can form the nucleus of syllable. One regularly encounters syllables of the shape *tK, rB, xZ, wL*, for example. (Capitalization represents nucleus-hood.) The following table provides illustrative examples, with periods used to mark syllable edges:4

| P&S’s notation for syllables translates into tree formalism like this: |
|---|---|---|
| σ | σ | σ |
| / | / | / |
| N | N | N |
| .rA.tK.tI |

“N” designates the Nucleus or head constituent of the syllable. Consonants before the nucleus in the syllable belong to the Onset (not shown); consonants after the nucleus belong to the Coda.

Pay attention to the boldfaced segments in each row of the following table. The first example, *rA.tK.tI*, shows that there are conditions where even a voiceless stop can be the nucleus of a syllable — rather unexpectedly, in view of our experience with more familiar languages. In English, for example, the only consonants that can head a syllable are the sonorants (=nasals and liquids), as in *bottLe, butteR, satiN, bottoM*.

Also pay attention to the consonants that *share* the syllable with the boldfaced nuclei. (For instance, the *L* of *tR.gLt* shares a syllables with *g* and *t*.) In general, where would those tautosyllabic neighbors be found on the chart? [NB: there are conditions under which this generalization is not respected perfectly.]

A point of notation. Nuclei are written as capital letters, and margins (non-nuclei) are written as lower-case letters. A slight departure from this notation involves writing *U/w* and *Y/i* as nuclear/non-nuclear counterparts. A further inconsistency: below, you’ll often see nuclear *i, u, a* written without capitalization.

---

4 Glosses are *ratki* ‘she will remember’; *bddl* ‘exchange!’; *maratgt* ‘what will happen to you? ’; *tftkt* ‘you suffered a strain’; *txzn* ‘you stored’; *txznak* ‘she even stockpiled’; *tznt* ‘it(f.)’ is stifling; *mzh* ‘she jested’; *trgt* ‘you locked’; *ildi* ‘he pulled’; *ratlalt* ‘you will be born’; *trba* ‘she carried-on-her-back’; where ‘you’ = 2psg. and English past translates the perfect.

Reports of native speaker intuitions or of impressionistic phonetic observations (i.e., listening) have some value in studies of syllabification, but they can’t really be relied on in tricky and complicated cases like this one. It’s important, then, that D&E (especially in their later papers) have found lots of additional evidence for how words are broken up into syllables and what the nuclei are. See the bibliography at the end of this document and the accompanying handout.

The domain of syllabification is the phonological phrase.

To say that the domain of syllabification is the phonological phrase is to say that syllables can cross word boundaries freely; presumably only outright pausing will stop syllabification. The ITB examples here are all words in isolation — they’re one-word phrases, so we really don’t need to worry about this stuff. (By the way, would you say that word boundaries affect syllabification in English? You might want to compare saw Ted with sought Ed or nitrate with night rate.)

All syllables must have onsets except when they occur in absolute phrase-initial position. There, syllables may begin with vowels, either with or without glottal striction (Dell & Elmedlaoui 1985: 127 fn. 20), evidently a matter of phonetic implementation.

In many languages, including ITB, syllables can’t begin with nuclei. Expressed positively, this is the requirement that every syllable start with an onset. There’s an exception, also found in many other languages besides ITB: initial syllables can be onsetless. More on this below. In ITB, apparently, onsetless initial syllables sometimes start with a /ʔ/ (the “glottal striction”), sometimes not. P&S are going to ignore the fleeting ʔ, regarding it as a phonetic matter, outside the purview of their phonological study.

Since any segment at all can form the nucleus of a syllable, there is massive potential ambiguity in syllabification, and even when the onset requirement is satisfied, a number of distinct syllabifications will often be potentially available. But the actual syllabification of any given string is almost always unique. Dell & Elmedlaoui discovered that assignment of nuclear status is determined by the relative sonority of the elements in the string. Thus we find the following typical contrasts:

<table>
<thead>
<tr>
<th>Nucleus Type</th>
<th>Example</th>
<th>Morphology</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>voiceless stop</td>
<td>rA.K.tl</td>
<td>ra-t-kti</td>
<td>1985: 113</td>
</tr>
<tr>
<td>voiced stop</td>
<td>bD.dL.</td>
<td>bddl</td>
<td>1988: 1</td>
</tr>
<tr>
<td></td>
<td>.ma.ra.Gt.</td>
<td>ma-ra-t-g-t</td>
<td>1985: 113</td>
</tr>
<tr>
<td>voiceless fricative</td>
<td>tF.tKt.</td>
<td>t-ftk-t</td>
<td>1985: 113</td>
</tr>
<tr>
<td></td>
<td>tX.zNt.</td>
<td>t-xzn-t</td>
<td>1985: 106</td>
</tr>
<tr>
<td>voiced fricative</td>
<td>txZ.nAkk*</td>
<td>t-xzn#nakk*</td>
<td>1985: 113</td>
</tr>
<tr>
<td>nasal</td>
<td>tzMt.</td>
<td>t-zmt</td>
<td>1985: 112</td>
</tr>
<tr>
<td></td>
<td>tM.zh.</td>
<td>t-mzh</td>
<td>1985: 112</td>
</tr>
<tr>
<td>liquid</td>
<td>tR.gLt.</td>
<td>t-rgl-t</td>
<td>1985: 106</td>
</tr>
<tr>
<td>high vowel</td>
<td>.l.l.d.</td>
<td>i-ldi</td>
<td>1985: 106</td>
</tr>
<tr>
<td></td>
<td>.rat.lUlt.</td>
<td>ra-t-lul-t</td>
<td>1985: 108</td>
</tr>
<tr>
<td>low vowel</td>
<td>tR.bA.</td>
<td>t-rba</td>
<td>1985: 106</td>
</tr>
</tbody>
</table>
Sonority Effects on Nuclear Status

(a) \( tzM \rightarrow ^* tzMt \) ‘\( m \) beats \( z \) as a nucleus’

(b) \( r\text{a.} Lt \rightarrow ^* r\text{a.} tL \text{ wL}. t \) ‘\( u \) beats \( l \) as a nucleus’

This stuff is what this chapter is all about — how do we determine which segments are going to be nuclei and which aren’t. D&E’s proposal is that sonority is the relevant property, with more sonorous segments being preferred as nuclei.

Sonority, or something like it, has been known since the time of Panini, the Sanskrit grammarian. The modern view of sonority derives from the work on Jespersen and others in the 19th Century. Physically, sonority is approximately correlated with relative openness of the vocal tract, greater airflow, and greater amplitude (more energy in the speech signal). The sonority scale, seen below in (2), is a pretty typical classification of speech sounds along this dimension: vowels are more sonorous than consonants, and low vowels are even more open, hence more sonorous, than high vowels. (Mid vowels are in the middle, of course.)

In syllables cross-linguistically, sonority usually rises at the beginning, reaches a peak at the nucleus, and falls off again at the end. Look at the following English syllables in this light:

- pit, lid, dill
- pat, lad, doll
- blue, bulb
- trill
- kiln
- but, spa, strong

All the structures in (3), including the ill-formed ones, are locally well-formed, composed of licit substructures. In particular, there is nothing wrong with syllables \( t\text{Z}, t\text{L}, \text{ or wL} \) nor with word-final sequences \( mt \) — but the more sonorous nucleus is chosen in each case. By examining the full range of such contrasts, Dell and Elmedlaoui establish the relevance of the following familiar kind of 8-point hierarchy:

(4) Sonority Scale

\[
\begin{align*}
| \text{Low V} | > | \text{High V} | > | \text{Liquid} | > | \text{Nasal} | > | \text{Voiced Fricative} | > | \text{Voiceless Fricative} | & > | \text{Voiceless Stop} | > \\
| \text{Voiceless Stop} | 
\end{align*}
\]

We write \( | \alpha | \) for the sonority or intrinsic prominence of \( \alpha \).

So now P&S are going to lay out D&E’s story. It’s a strongly operational one, to be replaced later by P&S’s constraint-based approach.

With the sonority scale in hand, Dell and Elmedlaoui then propose an iterative syllable-construction procedure that is designed to select the correct nuclei. Their algorithm can be stated as follows, modified slightly from Dell & Elmedlaoui 1985: 111(15):

(5) Dell–Elmedlaoui Algorithm for Core Syllabification (DEA)

Build a core syllable (“CV”) over each substring of the form XY, where

- X is any segment (except \( a \)), and
- Y is a matrix of features describing a step of the sonority scale.

Start Y at the top of the sonority scale and replace it successively with the matrix of features appropriate to the next lower step of the scale.

(Iterate from Left to Right for each fixing of the nuclear variable Y.)
We deal with the fact that /a/ cannot occupy syllable margins in §8.1.1. The commonly encountered relaxation of the onset requirement in initial position is resolved in McCarthy & Prince 1993 in terms of constraint interaction, preserving the generality of ONS. Dell & Elmedlaoui are themselves somewhat ambivalent about the need for directionality (Dell & Elmedlaoui 1985: 108); they suggest that “the requirement [of directionality] is not concerned with left to right ordering per se, but rather with favoring application of [the DEA] that maximize the sonority differences between [onset and nucleus]” (Dell & Elmedlaoui 1985: 127 fn. 22). In addition, they note that directionality falsely predicts *i.t.bd.rin. from /i=t-!bdri-n/ ‘for the cockroaches’, whereas the only licit syllabification is .it.bD.rin. The reason for this syllabification is not understood. A directionless theory leaves such cases open for further principles to decide.

Like all such procedures, the DEA is subject to the Free Element Condition (FEC: Prince 1986), which holds that rules establishing a level of prosodic structure apply only to elements that are not already supplied with the relevant structure. By the FEC, the positions analyzed by the terms X,Y must be free of syllabic affiliation. Effectively, this means that any element seized as an onset is no longer eligible to be a nucleus. By the FEC, then, the successive loops of analysis by the D&E algorithm can’t change syllable structure that’s already been imposed.

There are other syllabification phenomena in ITB that require additional rules beyond the DEA; we will abstract away from these and focus on the sense of DEA itself. We will also put aside some wrinkles in the DEA which are related to parenthesized expressions in (5) — the lack of a glide counterpart for /a/, and the phrase-initial loosening of the onset requirement, and the claimed left-to-rightness of the procedure.5

The DEA is a rule, or rather a schema for rules, of exactly the classical type A → B/C—D. Each rule generated by the schema has a Structural Description specified in featural terms and a Structural Change (‘construct a core syllable’). To see how it works, consider the following derivations:

You should work through these derivations. One caution: P&S have suppressed the inner loop of the algorithm, which enforces left-to-right directionality. As they note in fn. 6, the evidence for directionality is weak and inconsistent. (Directionality is only relevant when adjacent segments have identical sonority, and in fact the behavior of such sequences doesn’t seem consistent.) And as you’ll see below, the theory that P&S propose doesn’t capture that effect of the D&E algorithm. (So what about directionality in general? How does a constraint-based framework like OT handle that? We’ll be addressing this later in the term, but if you’re familiar with evidence for directional processes in phonology, you might want to think about the matter now.)

There’s a pun here. “DEA” = Dell-Elmedlaoui Algorithm or US Drug Enforcement Administration. Indeed, the phrase “DEA in Action” could be a newspaper headline.

(6) DEA in Action

---

5 We deal with the fact that /a/ cannot occupy syllable margins in §8.1.1. The commonly encountered relaxation of the onset requirement in initial position is resolved in McCarthy & Prince 1993 in terms of constraint interaction, preserving the generality of ONS. Dell & Elmedlaoui are themselves somewhat ambivalent about the need for directionality (Dell & Elmedlaoui 1985: 108); they suggest that “the requirement [of directionality] is not concerned with left to right ordering per se, but rather with favoring application of [the DEA] that maximize the sonority differences between [onset and nucleus]” (Dell & Elmedlaoui 1985: 127 fn. 22). In addition, they note that directionality falsely predicts *.i.t.bd.rin. from /i=t-!bdri-n/ ‘for the cockroaches’, whereas the only licit syllabification is .it.bD.rin. The reason for this syllabification is not understood. A directionless theory leaves such cases open for further principles to decide.
We show the form predicted by the DEA. The form is actually pronounced rAt.IUlt. because obstruents cannot be nuclear next to phrase boundaries, as mentioned in fn. 5.

<table>
<thead>
<tr>
<th>Steps of the DEA</th>
<th>/ratlult/ ‘you will be born’</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seek [X][+low,+syl] &amp; Build</td>
<td>(rA)tlt</td>
</tr>
<tr>
<td>Seek [X][-lo,+syl] &amp; Build</td>
<td>(rA)t(lU)lt</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,-nas]</td>
<td>–blocked by FEC–</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,+nas]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son, +cnt, +voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son, +cnt, -voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son,-cnt,+voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son,-cnt,-voi] &amp; Build</td>
<td>(rA)t(lU)(lT)6</td>
</tr>
</tbody>
</table>

(7) DEA in Action

<table>
<thead>
<tr>
<th>Steps of the DEA</th>
<th>/txznt/ ‘you sg.stored’</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seek [X][+low,+syl]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-lo,+syl]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,-nas] &amp; Build</td>
<td>tx(zN)t</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,+nas]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son, +cnt, +voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son, +cnt, -voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son,-cnt,+voi] &amp; Build</td>
<td>(tX)(zN)t</td>
</tr>
<tr>
<td>Seek [X][-son,-cnt,+voi]</td>
<td>–</td>
</tr>
<tr>
<td>Seek [X][-son,-cnt,-voi]</td>
<td>–</td>
</tr>
</tbody>
</table>

6 We show the form predicted by the DEA. The form is actually pronounced rAt.IUlt. because obstruents cannot be nuclear next to phrase boundaries, as mentioned in fn. 5.
Chapter 2

Prince & Smolensky

16

These are exactly the sort of questions that were fruitfully asked, for example, of the classic TG rule of Passive that moved subject and object, inserted auxiliaries, and formed a PP: why does the post-verbal NP move up not down? why does the subject NP move at all? why is by+NP a PP located in a PP position? and so on.

(8) DEA in action

<table>
<thead>
<tr>
<th>Steps of the DEA</th>
<th>/txznas/ ‘she stored for him’</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seek [X][+low,+syl] &amp; Build</td>
<td>txz(nA)s</td>
</tr>
<tr>
<td>Seek [X][-lo,+syl]</td>
<td>—</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,-nas]</td>
<td>-blocked by FEC -</td>
</tr>
<tr>
<td>Seek [X][-syl,+son,+nas]</td>
<td>—</td>
</tr>
<tr>
<td>Seek [X][-syl,+cnt,+voi] &amp; Build</td>
<td>t(xZ)(nA)s</td>
</tr>
<tr>
<td>Seek [X][-syl,+cnt,-voi]</td>
<td>—</td>
</tr>
<tr>
<td>Seek [X][-syl,-cnt,+voi]</td>
<td>—</td>
</tr>
<tr>
<td>Seek [X][-syl,-cnt,-voi]</td>
<td>—</td>
</tr>
</tbody>
</table>

DEA works, clearly. So what’s wrong with it? Here’s what P&S have to say:

The DEA provides an elegant and straightforward account of the selection of syllable nuclei in the language. But it suffers from the formal arbitrariness characteristic of re-writing rules when they are put to the task of dealing locally with problems that fall under general principles, particularly principles of output shape. (By ‘formal arbitrariness’, we mean that a formal system rich enough to allow expression of the desired rule will also allow expression of many undesired variations of the rule, so that the rule itself appears to be an arbitrary random choice among the universe of possibilities.) The key to the success of the DEA is the way that the variable Y scans the input, starting at the top of the sonority scale and descending it step by step as the iterative process unfolds. We must ask, why start at the top? why descend the scale? why not use it in some more elaborate or context-dependent fashion?\(^7\)

Let me try to make this criticism of DEA a little more concrete. Operational or rule-based theories like *SPE* (*The Sound Pattern of English* — Chomsky and Halle 1968) must, of course, tell you how rules apply to forms. In *SPE* in particular, the following proposals are made:

- The rules of a language consist of an ordered list: (R1, R2, R3, ...). R1 applies first, then R2 applies to the output of R1, the R3 to the output of R2. In general, no rule knows anything about what happened earlier in the derivation except insofar as the results of prior rule application are encoded in the form that is handed to it. (There’s no look-back or look-ahead — it’s often said that phonological derivations are “strictly Markovian”, after the Russian mathematician Markov who developed some important results in formal language theory.)

- Each rule applies to a form by identifying all the places where its conditions are met and then applying to them simultaneously.

The second point was always rather problematic, and much work in the immediate post-*SPE* era replaced it by a different idea: a rule applies to a form by iterating directionally across it. Start at the beginning or end, and work to the other side. You can see this in DEA, though as noted earlier it doesn’t play a very big role.

But no rule-based theory, prior to DEA, had ever proposed iterating through a scale — and that kind of iteration is the “heart” of DEA. Within assumptions that were fairly standard at the time, DEA’s idea of

---

\(^7\) These are exactly the sort of questions that were fruitfully asked, for example, of the classic TG rule of Passive that moved subject and object, inserted auxiliaries, and formed a PP: why does the post-verbal NP move up not down? why does the subject NP move at all? why is by+NP a PP located in a PP position? and so on.
scale-wise iteration represented a considerable innovation. And, in linguistic theory at least, innovation isn’t necessarily a good thing. You have to ask the question: are the options provided by this innovation actually attested in the world’s languages? That’s what P&S are doing, with specific challenges, in the paragraph immediately above.

The answers are to be found in the universal theory of syllable structure markedness. The more sonorous a segment it is, the more satisfactory it is as a nucleus. Conversely, a nucleus is more satisfactory to the degree that it contains a more sonorous segment. It is clear that the DEA is designed to produce syllables with optimal nuclei; to ensure that the syllables it forms are the most harmonic that are available, to use the term introduced in §1. Dell and Elmedlaoui clearly understand the role of sonority in choosing between competing analyses of a given input string; they write:

When a string …PQ… could conceivably be syllabified as …Pq… or as …pQ… (i.e. when either syllabification would involve only syllable types which, when taken individually, are possible in ITB), the only syllabification allowed by ITB is the one that takes as a syllabic peak the more sonorous of the two segments. (Dell & Elmedlaoui 1985:109)

But if phonology is couched in re-writing rules, this insight cannot be cashed in as part of the function that assigns structural analyses. It remains formally inert. Dell and Elmedlaoui refer to it as an “empirical observation,” emphasizing its extra-grammatical status.

The statement quoted from D&E is a particular instance of an important methodological point: often you will read something where the real insights are in the prose surrounding and justifying the analysis rather than in the analysis itself. In other words, there’s a serious disconnection between the D&E insight, just quoted, and the D&E analysis. The goal of a successful phonological theory is to bring the analysis into conformity with the insights.

The DEA itself makes no contact with any principles of well-formedness; it merely scans the input for certain specific configurations, and acts when it finds them. That it descends the sonority scale, for example, can have no formal explanation. But the insight behind the DEA can be made active if we re-conceive the process of syllabification as one of choosing the optimal output from among the possible analyses rather than algorithmic structure-building. ♦

“Choosing the optimal output from among the possible analyses” is a fundamental idea of OT. Universal Grammar (UG) provides an array of “possible analyses” that will be relevant in some empirical domain. For the case at hand, the relevant possible analyses will be different logically possible ways of syllabifying the same basic sequence of segments: /blt/ → .blt., .B.lT., .B.lT., .B.L.T., .bL.T., .bL.T., .bL.T., .blT., .blT., .blT., .blT., .blT., .blT., .blT., .blT., .blT., .blT. .... These are the candidates or the candidate set. And the optimal output will be the member of this set that best satisfies the conditions on syllabification in Berber — informally, no epenthesis (no empty nuclei), no deletion (no unsyllabified segments), good (=high sonority) nuclei. But I’m getting a bit ahead of things.

Let us first suppose, with Dell and Elmedlaoui, that the process of syllabification is serial, affecting one syllable at a time (thus, that it operates like Move-α or more exactly, Move-x of grid theory). ♦

Move-α and Move-x refer to two types of elementary operations or processes in syntax and phonology, respectively. Move-α is the syntactic movement transformation, and Move-x is the stress-movement transformation seen in the contrast between thiréen and thiréen mén. Because they are elementary operations, they can affect only one syntactic constituent or stress at a time; deriving an actual sentence or phonological phrase may involve many successive applications of these transformations.

At each stage of the process, let all possible single syllabic augmentations of the input be presented for evaluation. This set of candidates is evaluated by principles of syllable well-formedness and the most harmonic string in the set is selected as the output. We can state the process informally as follows:

(9) Serial Harmonic Syllabification (informal).
Form the optimal syllable in the domain.
Further development of this idea could eliminiate complications at the level of the general theory; in particular, the appearance of obeying the Free Element Condition during serial building of structure could be seen to follow from the fact that disobeying it inevitably decrements the harmony of the representation.

As an exercise for yourself, you might want to try pursue this conjecture.

### Explanation:
- **“At each stage of the process...”** The idea in this sentence is that each step or loop in the derivation involves taking the form as it’s currently syllabified and, while respecting all previous syllabification (the FEC), constructing various single new syllables on it. Suppose, for instance, that we’ve reached the point where we have the partially-syllabified form tx[zNt]. The candidates will include tx[zNt], T[x][zNt], [T][x][zNt], [T][X][zNt], and so on. A candidate like [T][X][zNt] isn’t legitimate, by P&S’s assumption in this sentence, because it isn’t a single syllabic augmentation; a candidate like t[xZ][Nt] isn’t legitimate because it isn’t a syllabic augmentation, it’s a syllabic transformation or change.
- **“This set of candidates...”** From the set of legitimate candidates, as just defined, the one with the best syllabification will be chosen. What’s best? Sonority principles are the determining factor.
- **“(9) Serial Harmonic Syllabification”.** This is an informal characterization of the algorithm just sketched — at each loop, just make the best syllable, and continue looping until you can no longer make any syllables.

This approach depends directly on the principles of well-formedness which define the notion ‘optimal’. No instructions are issued to the construction process to contemplate only one featurally-specified niche of the sonority scale. Indeed, the Harmonic syllabification algorithm has no access to any information at all about absolute sonority level or the specific featural composition of vowels, which are essential to the DEA; it needs to know whether segment α is more sonorous than segment β, not what their sonorities or features actually are. All possibilities are entertained simultaneously and the choice among them is made on grounds of general principle. That you start at the top of the scale, that you descend the scale rather than ascending it or touring it in some more interesting fashion, all this follows from the principles that define relative well-formedness of nucleus-segment pairings. The formal arbitrariness of the DEA syllable-constructing procedure disappears because the procedure itself (‘make a syllable’) has been stripped of intricacies.

The comments immediately above compare (9) to DEA. They are very important. Be sure you understand them, since understanding how to compare theories at this level is an essential skill.

This is an instance of harmony-increasing processing (Smolensky 1983, 1986; Goldsmith 1990, 1993). The general rubric is this:

(10) **Harmonic Processing**

Go to the most well-formed available state.

We conclude that the Dell-Elmedlaoui results establish clearly that harmonic processing is a grammatical mechanism; and that optimality-based analysis gives results in complex cases. Let us now establish a formal platform that can support this finding.

## 2.2 Optimality Theory
What, then, is the *optimal* syllable that Harmonic Syllabification seeks? In the core process that we are focussing on, two constraints are at play, one ensuring onsets, the other evaluating nuclei. The onset constraint can be stated like this:

(11) **The Onset Constraint (ONS).** Syllables must have onsets (except phrase initially).

As promised, we are not going to explicate the parenthesized caveat, which is not really part of the basic constraint (McCarthy & Prince 1993: §4).

One of the best results to emerge from OT is that stipulations like “(except phrase initially)” are best understood in terms of constraint interaction. Instead of saying “Syllables must have onsets (except phrase initially)”, the constraint ONS of UG really says “Syllables must have onsets” *tout court*. The glitch about phrase-initial syllables in Berber and some other languages is the result of satisfying another constraint, which mitigates the effect of ONS. For now, this is just a promissory note, but we’ll get to it soon.

The nuclear constraint looks like this:

(12) **The Nuclear Harmony Constraint (HNUC).** A higher sonority nucleus is more harmonic than one of lower sonority.

*I.e.* If \(|x| > |y|\) then Nuc/x ≻ Nuc/y.

The formalizing restatement appended to the constraint uses some notation that will prove useful.

For ‘x is more harmonic than y’ we write x ≻ y.

For ‘the intrinsic prominence of x’ we write \(|x|\).

The expression A/x means ‘x belongs to category A, x is the constituent-structure child of A’

The two kinds of order ≻ and > are distinguished notationally to emphasize their conceptual distinctness. Segments of high sonority are not more harmonic than those of lower sonority; it is only when segments are contemplated in a structural context that the issue of well-formedness arises.

What HNUC says, then, is this:

If the intrinsic prominence (=sonority) of x is greater than the intrinsic prominence of y, then a syllable with an x nucleus is more harmonic than a syllable with a y nucleus.

There are two distinct inequalities involved: x is greater than y in sonority and x-nucleus is more harmonic than y-nucleus. The constraint HNUC says, as part of UG, that there’s an implicational relation between these inequalities. It establishes a connection between a kind of physical quantity, the sonority scale, and an aspect of prosodic structure, the nuclei of syllables.

The statement of HNUC contains the phrase “more harmonic”; you should use this in place of the (seemingly hyperbolic) “more optimal”.

It is necessary to specify not only the relevant constraints, but also the set of candidates to be evaluated. To do this we need to spell out the function Gen that admits to candidacy a specific range of structurings or parses of the input. In the case at hand, we want something roughly like this:

(13) **Gen** (*input*): the set of (partial) syllabifications of *input*, which differ from input, in no more than one syllabic adjunction.

---

9 It is also possible to conceive of the operative constraint in a kind of ‘contrapositive’ fashion. Because all underlying segments of ITB are parsed, a segment is a nucleus iff it is not a member of the syllable margin. Consequently, negative constraints identifying the badness of syllable margins can have the same effect as positive constraints identifying the goodness of nuclei. We investigate this approach below in §8.1.1, §8.3.3, §8.4.2.
For the moment, we’re looking at an approach based on Serial Harmonic Syllabification. The “Serial” part is responsible for the restriction of Gen to “no more than one syllabic adjunction”. Here’s how things work, approximately (also see above, p. 18, comment on “At each stage of the process”:

(i) Inputs consist of strings of segments with no syllabic affiliations whatsoever: /abcd/

(ii) Gen takes the input and forms exactly one syllable (with nucleus designated). Hence, we get a candidate set like this: {[A] bcd, [Ab] cd, a[B] cd, [aB] cd, a[bC] d, ...} You get the idea. (Warning: I’ve ignored syllable-final consonants (=codas) in this simplified description of what Gen does.)

(iii) HNuc and ONS, interacting as described below, choose the best member of this set.

(iv) Now loop back to (ii), with the proviso that Gen won’t alter syllabification that’s already been imposed. (This is just the FEC, but cf. fn. 8).

(v) Quit when the derivation converges — that is, when the output of step \(n\) is no different from the output of step \(n-1\).

The serial treatment of syllabification in Berber, which is the reason for the loop just described, is very much a temporary expedient. P&S are trying to present first an approach that’s only minimally different from D&E’s, so they temporarily adopt D&E’s serialist assumptions. Those assumptions are going to be dropped before the chapter is out, however, and they’re not going to be very important in our work this semester.

For any form input, to undergo Serial Harmonic Syllabification, the candidate set Gen(input) must be evaluated with respect to the constraints ONS and HNuc. There would be little to say if evaluation were simply a matter of choosing the candidate that passes both constraints. Crucially, and typically, this straightforward approach cannot work. Conflict between the constraints ONS and HNuc is unavoidable; there are candidate sets in which no candidate satisfies both constraints.

“[T]here are candidate sets in which no candidate satisfies both constraints.” This is a vitally important point. Optimality Theory is about finding the best; it is not about achieving perfection. (Though it is sometimes wrongly thought to be — McCarthy and Prince 1994 dub this “the fallacy of perfection”.) For a given input, Gen will often supply no “perfect” candidate, and so the grammar of a particular language is obliged to select that member of the candidate set that is the best in relation to that particular grammar.

Consider, for example, the syllabification of of the form /haul-tn/ ‘make them (m.) plentiful’ (Dell & Elmedlaoui 1985:110). Both ONS and HNuc agree that the core syllable /A/ should be formed: it has an onset as well as the highest ranking nucleus. Similarly, we must have a final syllable /tN/. But what of the rest of the string?

That is, the discussion immediately below will focus only on the syllabification of the /ul/ substring of /haul-tn/.

The sound /h/ is a voiceless pharyngeal fricative. It patterns with the other voiceless fricatives (/f, s/) on the sonority hierarchy.

We have two choices for the sequence /ul/: a superior nucleus lacking an onset, as in /u/, or an onsetted syllable with an inferior nucleus, as in /wL/. (We write \(u\) for the nuclear version, \(w\) for the marginal version of the high back vocoid.) This situation can be perspicuously displayed in tabular form:

(14) Constraint Inconsistency

---

10 Properly speaking, if we limit our attention to the core syllable stage of the procedure, we should be comparing core /u/ with core /wL/. But the comparison remains valid even after coda consonants are adjoined and we wish to emphasize that the two cited analyses of /haul-tn/ differ only in treatment of the sequence /ul/.
The cells contain information about how each candidate fares on the relevant constraint. A blank cell indicates that the constraint is satisfied; a star indicates violation. (In the case of a scalar constraint like HNUC we mention the contents of the evaluated element.) The first form succeeds on ONS, while the second form violates the constraint. The relative performance is exactly the opposite on HNUC: because $|u| > |l|$, the second, onsetless form has the better nucleus. The actual output is, of course, /haul-tN/. The onset requirement, in short, takes priority.

You might find it helpful to translate the HNUC column into *’s, especially if you’ve encountered OT before. In that case, what you want is a way of stating HNUC that directly indicates degree of deviation from the highest step on the sonority scale. Here’s a suggestion:

**HNUC (Revised)**

If Nuc/x, then assign one * for each step in $a – x$.

The table then reads:

<table>
<thead>
<tr>
<th>Candidates</th>
<th>ONS</th>
<th>HNUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>/haul-tN/</td>
<td></td>
<td></td>
</tr>
<tr>
<td><del>.wL.</del></td>
<td></td>
<td>**</td>
</tr>
<tr>
<td><del>.U1.</del></td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

The best candidate with respect to a given constraint is just the one that has the fewest *’s.

Such conflict is ubiquitous, and to deal with it, we propose that a relation of dominance, or priority-ranking, can be specified to hold between constraints. When we say that one constraint dominates another, we mean that when they disagree on the relative status of a pair of candidates, the dominating constraint makes the decision. If the dominating constraint does not decide between the candidates — as when both satisfy or both violate the constraint equally — then the comparison is passed to the subordinate constraint. (In the case of a more extensive hierarchy, the same method of evaluation can be applied repeatedly.)

Dominance, and with it violation, are further fundamental ideas, essential to the success of the enterprise. Simple satisfaction of output constraints just isn’t enough to define grammars and capture all the internal and cross-linguistic generalizations. Many significant phonological insights involve tendencies or preferences rather than absolutes. Domination of one constraint by another is a (remarkably successful) proposal about how to formalize these tendencies or preferences and use them to define grammars with none of the muzziness or wishy-washyness that terms like “tendency” normally evoke.

In the case at hand, it is clear that ONS must dominate HNUC. The top priority is to provide syllables with onsets; the relative harmony of nuclei is a subordinate concern whose force is felt only when the ONS issue is out of the way. ➔

In fact, ONS (as stated in (11), though not as we ultimately understand it), is always obeyed in ITB. It is, in Optimality-Theoretic parlance, an undominated and therefore unviolated constraint.

We will write this relation as ONS >> HNUC. Given such a hierarchy, an optimality calculation can be usefully presented in an augmented version of display (14) that we will call a constraint tableau:

(15) **Constraint Tableau** for partial comparison of candidates from /haultn/
Chapter 2

Princeton University Press

Candidates | ONS | HNUC
---|---|---
\( \varepsilon \varepsilon \sim .wL.\sim \) | | \( |1| \)
\( \sim .uL.\sim \) | * ! | \( |u| \)

Constraints are arrayed across the top of the tableau in domination order. As above, constraint violations are recorded with the mark *, and blankness indicates total success on the constraint. These are the theoretically important conventions; in addition, there is some clarificatory typography. The symbol \( \varepsilon \varepsilon \) draws the eye to the optimal candidate; the ! marks the crucial failure for each suboptimal candidate, the exact point where it loses out to other candidates. Cells that do not participate in the decision are shaded. In the case at hand, the contest is decided by the dominant constraint ONS; HNUC plays no role in the comparison of .wL. and .uL. HNUC is literally irrelevant to this particular evaluation, as a consequence of its dominated position — and to emphasize this, we shade its cells. Of course, HNUC is not irrelevant to the analysis of every input; but a precondition for relevance is that there be a set of candidates that all tie on ONS, all passing or all failing it.

The role of the constraint tableau in OT is similar to the role of the derivation in SPE or the truth table in logic: it’s the essential tool for proving and displaying the results of the analysis. You are urged to use tableaux frequently and liberally in your work. You are also urged to begin the analysis with simple 2X2 tableaux like (15) — two constraints and two candidates, one a winner and one a loser — only gradually assembling the results of various 2X2 tableaux into more complex pictures.

If we were to reverse the domination ranking of the two constraints, the predicted outcome would be changed: now .uL. would be superior to .wL. by virtue of its relative success on HNUC, and the ONS criterion would be submerged. Because of this, the ranking ONS \( \gg \) HNUC is crucial; it must obtain in the grammar of Berber if the actual language is to be generated.

Latent in the paragraph immediately above is another core idea of OT: permuted ranking or factorial typology. The constraints are given by universal grammar (UG); the ranking is particular to individual languages. Thus, any permutation of the constraints of UG should be a possible human language, if OT and the hypothesized constraints of UG are correct.

The notion of dominance shows up from time to time in one form or another in the literature, sometimes informally, sometimes as a clause clarifying how a set of constraints is to be interpreted. For example, Dell and Elmedlaoui write, “The prohibition of hiatus...overrides” the nuclear sonority comparison (Dell & Elmedlaoui 1985: 109, emphasis added). For them, this is an extra-grammatical observation, with the real work done by the Structural Descriptions provided by the DEA and the ordering of application of the subrules. Obviously, though, the insight is clearly present. Our claim is that the notion of dominance, or ‘over-riding’, is the truly fundamental one. It is the mechanisms involved in constructing elaborately specific Structural Descriptions and modes of rule application that deserve extra-grammatical status.

To see how Serial Harmonic Syllabification (9) proceeds, let us examine the first stage of syllabifying the input /txznt/ ‘you sg. stored, pf.’. It is evident that the first syllable constructed must be .zN. — it has an onset, and has the highest sonority nucleus available, so no competing candidate can surpass or even equal it. A more discursive examination of possibilities might be valuable; the larger-scale comparison are laid out in the constraint tableau below.

Here are (some of the) leading candidates in the first round of the process:
(16) **Constraint Tableau for Serial Syllabification** of /txzn/ (partial, first step)

<table>
<thead>
<tr>
<th>Candidates</th>
<th>ONS</th>
<th>HNUC</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>tx(zN)t</td>
<td>n</td>
<td>n</td>
<td>optimal: onsetted, best available nucleus</td>
</tr>
<tr>
<td>txz(N)t</td>
<td>* !</td>
<td>n</td>
<td>no onset, HNUC irrelevant</td>
</tr>
<tr>
<td>t(xZ)nt</td>
<td>z !</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(tX)znt</td>
<td>x !</td>
<td></td>
<td></td>
</tr>
<tr>
<td>txz(nT)</td>
<td>t !</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Some suggestions for practice with the tools:

(i) Rewrite (16) so that HNUC violations are indicated by *’s; the best candidate with respect to HNUC then has the fewest *’s.

(ii) Invent a few additional candidates and show how they are evaluated by (16).

(iii) Construct a tableau for the next step of serial harmonic syllabification, taking tx(zN)t as input. (For now, we’re not worrying about the coda, so you should ignore the final t.)

(iv) Pick some other Berber words from earlier in the chapter and do the same thing for them.

Syllabic parsing is conceived here as a step-by-step serial process, just as in the DEA. A candidate set is generated, each produced by a single licit change from the input; the relative status of the candidates is evaluated, yielding an optimal candidate (the output of the first step); and that output will then be subject to a variety of further single changes, generating a new candidate set to be evaluated; and so on, until there are no bettering changes to be made: the final output has then been determined.

This step-by-step Harmony evaluation is not intrinsic to the method of evaluation, though, and, in the more general context, it proves necessary to extend the procedure so that it is capable of evaluating entire parsed strings, and not just single (new) units of analysis. To do this, we apply the same sort of reasoning used to define domination, but within the constraint categories. To proceed by example, consider the analysis of /txzn/ taking for candidates all syllabified strings. We present a sampling of the candidate space. As I noted above, the serial assumption isn’t maintained for long. It isn’t at all necessary, at least in the case of Berber.

(17) **Parallel Analysis of Complete Syllabification** of /txzn/

<table>
<thead>
<tr>
<th>Candidates</th>
<th>ONS</th>
<th>HNUC</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>.tX.zNt</td>
<td>n</td>
<td>x</td>
<td>optimal</td>
</tr>
<tr>
<td>.Tx.zNt</td>
<td>n</td>
<td>t !</td>
<td></td>
</tr>
<tr>
<td>.tXZ.nT</td>
<td>x !</td>
<td>t</td>
<td></td>
</tr>
<tr>
<td>.txZ.Nt</td>
<td>* !</td>
<td>z</td>
<td></td>
</tr>
<tr>
<td>.T.X.Z.N.T</td>
<td>* !</td>
<td>* ***</td>
<td></td>
</tr>
</tbody>
</table>

In evaluating the candidates we have kept to the specific assumptions mentioned above: the onset requirement is suspended phrase-initially, and the nonnuclear status of peripheral obstruents is, as in the DEA itself, put aside (see fn. 5).
In this tableau, all the relevant information for harmonic evaluation of the parse of the whole string is present. We start by examining the first column, corresponding to the dominant constraint ONS. Only the candidates which fare best by this constraint survive for further consideration. The first three candidates all have syllables with onsets; the last two do not (to varying degrees). Lack of onset in even a single non-initial syllable is immediately fatal, because of the competing candidates which satisfy ONS. The remaining three parses are not distinguished by ONS, and so HNUC, the next constraint down the hierarchy, becomes relevant. These three parses are compared by HNUC as follows. The most sonorous nucleus of each parse is examined: these are the most harmonic nuclei according to HNUC. For each of the first two candidates the most sonorous nucleus is $n$. For the last candidate, the most sonorous nucleus is $x$, and it drops out of the competition since $n$ is more sonorous than $x$. We are left with the first two candidates, so far tied on all comparisons. The HNUC evaluation continues now to the next-most-harmonic nuclei, where the competition is finally settled in favor of the first candidate $tX.zNt$.

To understand more fully how HNUC works here, you might find it helpful to translate the violations into *'s, by the procedure I described earlier. Then you can simply lump together all the HNUC *'s in each row. For instance,

- $tX.zNt$ gets *** for $N$ and ***** for $x$, totaling ********
- while $Tx.zNt$ gets *** for $N$ and ******* for $T$, totaling **********.

So $tX.zNt$ fares better with respect to this constraint.

What we have done, in essence, is to replace the iterative procedure (act/evaluate, act/evaluate,…) with a recursive scheme: collect the results of all possible actions, then sort recursively. Rather than producing and pruning a candidate set at each step of sequential process, striving to select at each step the action which will take us eventually to the correct output, the whole set of possible parses is defined and harmonically evaluated. The correct output is the candidate whose complete structure best satisfies the constraint hierarchy. And ‘best satisfies’ can be recursively defined by descending the hierarchy, discarding all but the best possibilities according to each constraint before moving on to consider lower-ranked constraints.

The great majority of analyses presented in this paper will use the parallel method of evaluation. A distinctive prediction of the parallel approach is that there can be significant interactions of the top-down variety between aspects of structure that are present in the final parse. In §4 and §7 we will see a number of cases where this is borne out, so that parallelism is demonstrably crucial; further evidence is presented in McCarthy & Prince 1993. ‘Harmonic serialism’ is worthy of exploration as well, and many hybrid theories can and should be imagined; but we will have little more to say about it. (But see footnote 107 below on Berber syllabification.)

“Significant interactions of the top-down variety” means that the structure of constituent A can influence the structure of constituent B, even though A is higher up in the tree than B. We’ll be discussing concrete examples of this later in the term. Serial theories build constituents bottom-up, so they encounter difficulties with top-down effects. But if A and B are built simultaneously, because complete, fully-formed output candidates are evaluated in parallel, then top-down and bottom-up effects are equally expected.

The notion of parallel analysis of complete parses in the discussion of constraint tableau (17) is the crucial technical idea on which many of our arguments will rest. It is a means for determining the relative Harmonies of entire candidate parses from a set of conflicting constraints. This technique has some subtleties, and is subject to a number of variant developments, so it is worth setting out with some formal precision exactly what we have in mind. A certain level of complexity arises because there are two dimensions of structure to keep track of. On the one hand, each individual constraint typically applies to several substructures in any complete parse, generating a set of evaluations. (ONS, for example, examines every syllable, and there are often several of them to examine.) On the other hand, every grammar has multiple constraints, generating multiple sets of evaluations. Regulating the way these two dimensions of multiplicity interact is a key theoretical commitment.
Our proposal is that evaluation proceeds by constraint. In the case of the mini-grammar of ONS and HNUC, entire syllabifications are first compared via ONS alone, which examines each syllable for an onset; should this fail to decide the matter, the entire syllabifications are compared via HNUC alone, which examines each syllable’s nucleus. That’s why I suggested above that you lump together all the HNUC *’s. All the violation-marks incurred by a form with respect to any given constraint are the same. We don’t try to “match” candidates syllable for syllable, as the following paragraph describes:

Another way to use the two constraints would be to examine each (completely parsed) candidate syllable-by-syllable, assessing each syllable on the basis of the syllabic mini-grammar. The fact that ONS dominates HNUC would then manifest itself in the harmony assessment of each individual syllable. This is also the approach most closely tied to continuous Harmony evaluation during a step-by-step constructive derivation. Here again, we do not wish to dismiss this conception, which is surely worthy of development. Crucially, however, this is not how Harmony evaluation works in the present conception.

In order to characterize harmonic comparison of candidate parses with full generality and clarity, we need to specify two things: first, a means of comparing entire candidates on the basis of a single constraint; then, a means of combining the evaluation of these constraints. The result is a general definition of \textit{Harmonic Ordering of Forms}; this is, in its formal essence, our theory of constraint interaction in generative grammar. It is the main topic of §5.

\section*{2.3 Summary of discussion to date}

The core syllabification of Imdlawn Tashliyt Berber provides a particularly clear case where the function assigning structural analyses must be based on the optimality of the output if it is to be properly founded on principle. Once the relevant principles have been moved into grammatical theory, it becomes possible to undertake a radical simplification of the generative procedure that admits (candidate) syllable structures. The focus shifts away from the effort to construct an algorithm that assembles the correct structure piece-by-piece, an effort that we believe is doomed to severe explanatory shortcomings. Linguistic theory, properly conceived, simply has little to say about such constructional algorithms, which (we claim) are no more than implementations of grammatical results in a particular computational-like framework. The main explanatory burden falls the constraints themselves, and on the apparatus that governs their interactions.

The Berber situation is particularly interesting in that core syllabification simply cannot proceed without the intervention of \textit{two} distinct constraints. As with other forms of prosodic organization, the most common picture is one in which the structure is built (more-or-less) bottom-up, step-by-single-step, with each step falling under the constraints appropriate to it. Taking this seriously in the syllable structure domain, this would mean, following Levin [Blevins] (1985) and ultimately Kahn (1976), that you first locate the nuclei — the heads of syllables; then project higher order structure that includes the onsets; then project the structure that includes postnuclear consonantism. In ITB, however, as in many other languages, the availability of nuclei depends on the choice of onsets: an early step in the derivational constructive procedure, working on a low level in the structural hierarchy, depends on later steps that deal with the higher levels. Indeed, the higher level constraint is very much the more forceful. Technical solutions to this conundrum can be found in individual cases — Dell and Elmedlaoui’s being a particularly clever one; but the theme will re-appear persistently in every domain of prosody, defying a uniform treatment in constructionist terms.

The point in the paragraph immediately above is a nice one, very much worth thinking about. In the theory advocated here, where outputs are evaluated, we expect exactly this kind of interaction. The whole output is open to inspection; how we choose to inspect it, or how we are forced by UG to inspect it, is not determined by the course that would be taken in bottom-up construction. The potential force of a
constraint is not indexed to the level of structure that it pertains to, and under certain circumstances (UG permitting or demanding), constraint dominance will be invoked to give higher-level constraints absolute priority over those relevant to the design of lower structural levels.

References
http://www.phon.ox.ac.uk/~jcoleman/berber.html has recorded specimens of Berber that you can play if you have a sound card in your computer.