Sonority-Driven Derivation of Syllable Structure

A generalization of Dell and Elmedlaoui’s Syllabification algorithm

The influential templatic theory of syllabification proposed by Ito (1986, 1989) has been successful for a wide variety of languages. But there are languages which resist a templatic account. The Imdlawn Tashlihiyt dialect of Berber (ITB), well-known through studies of Dell and Elmedlaoui (1985), is one such language. Dell and Elmedlaoui (henceforth D&E) proposed an account of ITB which is based on a sonority controlled iterative process of what they termed core syllable construction. Their account is widely viewed as some kind of special exception to the general rule that syllabification is essentially templatic.

The thesis of this paper is that D&E’s account of ITB syllabification is not exceptional. Suitably modified, D&E’s proposal is the general one; syllable templates are epiphenomenal, derived from local interactions. A convincing demonstration of this thesis is too large a project for a modest paper. This paper is intended only to provide some evidence for it. D&E’s analysis of ITB is first generalized and then applied in a straightforward way to two other languages which have syllable building processes which differ significantly from the ITB process and from each other. The first is syllabification in Khalka Mongolian (KM), which Svantesson (1995) shows can be compactly analyzed as directional iteration of maximal template matching. ITB is famous for the fact that any phoneme is permitted to be nuclear while in KM only vowels can be nuclear. ITB makes no use of epenthesis in building syllable structure while KM makes extensive use of epenthesis to build syllable structure. The second is Ath-Sidar Rifian Berber, which was analyzed by Dell and Tangi (1993). It shares some of the characteristics of KM, with extensive use of epenthesis. In spite of the differences, it will be demonstrated that at a suitable level of abstraction the mechanism of syllable building is essentially the same in Imdlawn Tashlihiyt Berber, Khalka Mongolian, and Ath-Sidar Rifian Berber. The differences come mainly from different Sonority Hierarchies and different conditions on which phonemes can and cannot be nuclei.

1. Dell and Elmedlaoui’s analysis of Imdlawn Tashlihiyt Berber syllabification

Some examples of the data which D&E explain are given in (1). In surface forms, syllable nuclei are indicated by blodface and nonnuclear i and u are represented by y and w. Remarkably, any phoneme can be a syllabus nucleus. k is a nucleus in (1a,b), f is a nucleus in (1b), and x is a nucleus in (1f).

(1) underlying surface gloss
a. ra-t-kti ra.tk.ti ‘she will remember’
b. t-ftk-t tf.tkt ‘she suffered a sprain’
c. ugl-x-tnt u.glx.tnt ‘I hung them (f.)’
d. haul-tn ha.wl.tn ‘make them (m.) plentiful’
e. rgl-x r.glx ‘I locked’
f. sx:n sx.xn ‘dip (in sauce)’

* I would like to thank Morris Halle for a very careful reading of an earlier draft; François Dell and Mohamed Elmedlaoui for helpful comments; and Sam Gutmann for extensive feedback on the ideas.
The syllable structure of (1f) is:

\[
\sigma_s \sigma_n
\]

This makes it clear that it is inaccurate to say that “x is a nucleus”. We could equally well have said “x is an onset”. It is not x that is a nucleus, but the timing slot associated with x. Syllables are groups of timing slots associated with an abstract syllable element, not groups of phonemes. The usual terminology should be understood in this way. The statement “any phoneme can be a syllable nucleus” should be understood as “any phoneme can be associated with a nuclear timing slot.”

D&E divide syllabification into two processes, an iterative rule of core syllabification followed by an iterative rule which adjoins the timing slots which are still unsyllabified as codas if possible, otherwise as onsets. Core syllabification is carried out by applying the rule schema (2b) iteratively (in a manner specified below) subject to a derivational constraint against hiatus (syllables with adjacent nuclei).

\[
(2) \quad \begin{align*}
\text{a. Project Doublet:} & \quad \times' \times \rightarrow \times' \times \\
\text{b. Project Singlet:} & \quad \times \rightarrow \times
\end{align*}
\]

\[
R = \begin{bmatrix}
\text{Project Doublet} \\
\text{Project Singlet}
\end{bmatrix}
\]

Both × and ×′ are assumed to be unsyllabified. The schema is understood to apply to the timing slot which becomes the nucleus, so Project Doublet applies to × in (2a), not ×′.

D&E’s major insight was to realize that the key to core syllabification was to target at each step the unsyllabified timing slot with maximal sonority among the potential targets of \(R\). A derivational constraint against hiatus is imposed on the application of \(R\), so the potential targets consists of the set of timing slots which the constraint against hiatus does not prevent \(R\) from applying to. Maximal sonority is determined with respect to a fine-grained sonority scale, which D&E take to be a partition of the phoneme inventory into an ordered list of disjoint sets of phonemes. The partition below is partial, restricted to phonemes which appear in the examples in this paper. Early in the list corresponds to higher ranking.

\[
\text{(3) ITB Sonority scale}
\]

<table>
<thead>
<tr>
<th>phonemes</th>
</tr>
</thead>
<tbody>
<tr>
<td>low vowel</td>
</tr>
<tr>
<td>high vowel</td>
</tr>
<tr>
<td>liquid</td>
</tr>
<tr>
<td>nasal</td>
</tr>
<tr>
<td>voiced fricative</td>
</tr>
<tr>
<td>voiceless fricative</td>
</tr>
<tr>
<td>voiced stop</td>
</tr>
<tr>
<td>voiceless stop</td>
</tr>
</tbody>
</table>

If (4) is applied iteratively to the leftmost potential target of maximal sonority, the core syllable structure of the examples (1) is produced.
Derivational constraints on the application of a rule schema will be written as above, with the constraints following the rule, separated by a semicolon.

The derivations of the examples in (1) are given in (5). The potential targets are boxed and their sonority maxima are shaded.

Note that (5b) requires left to right application. Otherwise the final $t$ could be targeted at the last step instead of the pre-final $k$, incorrectly producing $tt\.kn$. Another example of the need to stipulate left to right iteration at each sonority level is provided by /bain-n/ ‘they (m.) appear’, which surfaces as $ba\.yn$, not $bay\.nn$.

Some refinements are needed to account for the full range of ITB words. The analysis presented to this point makes incorrect predictions for the examples (6).

In order to account for (6a–d), D&E impose some additional derivational constraints on the syllable building rules. They both constrain the internal structure of syllables, as opposed to *Hiatus, which constrains the relations between syllables. The are 1) obstruents cannot project onsetless syllables, and 2) the second timing slot of a geminate cannot project a syllable. Initial obstruents and the second timing slot of geminates are thereby removed from the set of potential targets. This move correctly accounts for (6a–d), as shown in (7). We will return to consider (6e) shortly.
Example (6e) is not the result of conditions on the internal structure of syllables but of an exception to the choice of the leftmost sonority maximum at each step. If the initial and post-initial timing slots are both sonority maxima, the post-initial timing slot is chosen, presumably to avoid an onsetless syllable. So:

(8)  
\[
\sigma \\
\sigma \\
\sigma \\
\sigma
\]

D&E suggest that the second timing slot of a geminate cannot project a syllable because it is actually a bare timing slot at the point that syllabification takes place and is not associated with the phoneme of the slot to its left until after syllabification takes place. We follow this suggestion and take (9) to be the ITB *Syllable Structure Constraints (SSC)*.

(9)  
\[
\begin{align*}
\text{a. Obstruents cannot project onsetless syllables.} \\
\text{b. A bare timing slot cannot project a syllable.}
\end{align*}
\]

Formally, SSC is incorporated into the theory as a derivational constraint on syllable building. In place of (4), syllabification is carried out by iterated application of (10).

(10)  
\[
\mathcal{R} = \left[ \begin{array}{c}
\text{Project Doublet} \\
\text{Project Singlet}
\end{array} \right] : \left\{ \text{SSC} \} \right. \left\{ \text{Hiatus} \}
\]

2. The Generalized Dell and Elmedlaoui Algorithm (GDE)

We first leave the ITB rule \( \mathcal{R} \) (10) unchanged and concentrate on the way it iterates. In the formulation which ended the last section, various disparate factors combined to specify the target of \( \mathcal{R} \) at each step: the sonority relation, directionality, and an overrule of directionality at the left edge under certain conditions. These different factors can be combined into a single relation, which I will call (Syllabic) Prominence, a refinement of the Sonority relation in which positional factors play a secondary role.

(11)  
\[
\begin{align*}
\text{Prominence} &= \left[ \begin{array}{c}
\text{Sonority} \\
\text{Noninitial} \\
\text{Left}
\end{array} \right]
\end{align*}
\]

The schema (an ordered list in this case) of relations is interpreted as a relation in the usual way: \( \times \) is more prominent than \( \times' \) if it is more sonorous than \( \times' \) (i.e. it is associated with a phoneme that is more sonorous than the phoneme associated with \( \times' \)), or they are equally sonorant but \( \times \) is noninitial and \( \times' \) is not, or they are equally sonorant and neither is initial, but \( \times \) is to the left of \( \times' \).
The distinction between prominence and sonority is illustrated in the examples in (12). The top row gives the underlying and surface forms, the other rows shows the sonority and prominence relation between adjacent slots in the underlying form.

(12) a. txznas/txznas  
   Son.  t < x < z < n < a > s  
   Prom. t < x < z < n < a > s  

b. tftkt/tftekt  
   Son.  t < f > t = k = t  
   Prom. t < f > t > k > t  

c. iulm/yulm  
   Son.  i = u > l > m  
   Prom. i < u > l > m  

d. ldiyi/ldiyi  
   Son.  l > d < i = i > x < i  
   Prom. l > d < i > i > x < i  

The iteration scheme (13) yields the desired core syllable structure.

(13) Apply $R$ recursively to the local prominence maxima in the set of potential targets of $R$.

$\times$ is a local prominence maximum in the set of potential targets of $R$ if there is no timing slot in the set of potential targets which has greater sonority than $\times$ and is adjacent to $\times$. The set of potential targets consists of all the unsyllabified timing slots which the derivational constraints do not prevent $R$ from applying to; i.e. those that do not follow a nucleus, and are neither initial position obstruents nor bare (the second slot of surface geminates).

The order in which local prominence maxima are targeted is not specified by (13). For ITB the same results are produced if local prominence maxima are chosen at random or if $R$ is applied simultaneously to all of the local prominence maxima. Note that local prominence maxima can never be adjacent because directionality ensures that two timing slots can never be equally prominent, so application of $R$ at two different local prominence maxima cannot interfere with each other. There is evidence from other languages which favors simultaneous application, which is what I will assume. Although I will leave systematic discussion for future work, the idea is that simultaneous application gives a way to explain the typical sonority profiles of complex codas and complex onsets.

Consider, for example, the syllabification of an underlying form like tatnta in a language which allows complex onsets. Suppose also that $n$ has higher prominence than $t$. Serial application of $R$ could produce:

\[
\begin{array}{cccccccc}
  t & a & t & n & t & a & \rightarrow & \sigma \\
  t & a & t & n & t & a & \rightarrow & \sigma \\
  t & a & t & n & t & a & \rightarrow & \sigma \\
  t & a & t & n & t & a & \rightarrow & \sigma \\
  t & a & t & n & t & a & \rightarrow & \sigma \\
\end{array}
\]

The problem here is the sonority profile of the onset of the second syllable, with the initial consonant of the complex onset having higher sonority than the prenuclear consonant. Simultaneous application, on the other hand, forces syllable building at all of the growth loci of:

\[
\begin{array}{cccccccc}
  t & a & t & n & t & a
\end{array}
\]

If languages are impelled to form $R$ so that it applies to all growth loci (except perhaps at the edges, where unsyllabified elements are permitted in some languages), then $n$ cannot form a complex onset with the following $t$. Languages have options. Epenthesis might be a possibility, or $n$ simply deleted (the case in Icelandic, for example).

The examples in (14) illustrate how (12) and (13) combine to produce the desired core syllable structure. To make it easier for the reader to follow, the set of potential targets is boxed, the relative prominence between potential targets shown, and the local prominence maxima are shaded.
The next step in generalizing D&E’s algorithm is to expand the repertoire of elementary syllable building rules to include Adjoin Coda and Adjoin Onset (with obvious descriptions) which can apply to unsyllabified timing slots. Additionally, suppose that Project Doublet can steal a coda slot from an adjacent syllable. The following is permitted, for example.

In the second step, the slot associated with \( r \) disassociates from the first syllable when the second syllable is projected.

Say that an unsyllabified timing slot is a growth locus if there is no adjacent unsyllabified timing slot with greater prominence. Then ITB is specified by the Prominence relation (which includes the Sonority Hierarchy), the Syllable Structure Constraints, and \( \mathcal{R} \) below, which applies iteratively to the growth loci.

Illustrative derivations are given below.

(17) a. \( b < a > i > n > n \) → \( b \overset{\sigma}{a} i \overset{\sigma}{n} \rightarrow b \overset{\sigma}{a} i \overset{\sigma}{n} n \rightarrow b \overset{\sigma}{a} i \overset{\sigma}{n} n \rightarrow b \overset{\sigma}{a} i \overset{\sigma}{n} n \)

b. \( d > i > i > x \rightarrow \overset{\sigma}{d} i \overset{\sigma}{i} \times \overset{\sigma}{i} \rightarrow \overset{\sigma}{d} i \overset{\sigma}{i} \times \overset{\sigma}{i} \)}
Note that the order of Adjoin Coda before Adjoin Onset is required in the last step of (17b) and (17e).

The final step in transforming D&E’s algorithm for ITB syllabification into a general form is to reorganize the constraints under which syllabification takes place. The strict derivational constraint *Hiatus will be replaced by a pair of avoidance constraints, *No Onset and *Complex Onset.  

Consider the rule (18) where strict derivational constraints follow the semicolon and avoidance constraints, an ordered schema in this case, follow “&”.

\[
\begin{align*}
    18. \quad R &= \begin{bmatrix}
        \text{Project Syllable} \\
        \text{Adjoin Onset} \\
        \text{Adjoin Coda}
    \end{bmatrix}; \text{SSC} \& \begin{bmatrix}
        \text{*Complex Onset} \\
        \text{*No Onset}
    \end{bmatrix}
\end{align*}
\]

Syllabification is accomplished by applying \( R \) recursively to growth loci. The significance of the ordering of the two avoidance constraints is that onsetless syllables cannot be avoided at the expense of creating a syllable with a complex onset. The avoidance constraints take precedence over rule order, so rule order is relevant only when there is more than one rule whose application minimally violates the avoidance constraints.

The examples below illustrate the effect of the avoidance constraints. In (19a), *No Onset forces application of the lower ranked Adjoin Coda. In (19b), a *No Onset violation is preferable to the only alternative, since the alternative would result in a *Complex Coda violation.

1. The terminology is different from that of Halle and Idsardi (1995), who used the term avoidance constraint to mean a derivational constraint. My terminology agrees with Chomsky’s (1981) use in the Avoid Pronoun constraint, which imposed “a choice of PRO over an overt pronoun where possible”, where the italics are mine.
Note also that the first steps (17b) and (17e) violate *No Onset, but there is no way to avoid it.

3. Khalka Mongolian syllabification

I rely on the particularly clear analysis of KM syllabification by Svantesson (1995). Some examples of the data which must be accounted for are given in (20). The KM vowels are \{i, e, u, o, a, æ\} and the epenthetic vowel æ.

(20)  
<table>
<thead>
<tr>
<th>underlying</th>
<th>surface$^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. bolžmr</td>
<td>bolž.mar</td>
</tr>
<tr>
<td>b. zims-t</td>
<td>zimst</td>
</tr>
<tr>
<td>c. tarwq</td>
<td>tar.wøq</td>
</tr>
<tr>
<td>d. jornc</td>
<td>jor.tøn</td>
</tr>
<tr>
<td>e. zowl-lo</td>
<td>ze.wøl.lo</td>
</tr>
<tr>
<td>f. gorldai</td>
<td>gor.cøl.dai</td>
</tr>
</tbody>
</table>

An epenthetic vowel æ appears in all but (20b). The main thing that needs to be accounted for is the position of the epenthetic vowels. Why not jort.nc or zim.sot?

Svantesson shows that the partition of the phoneme inventory into the four sonority classes (21) is crucial in understanding KM syllabification. Here, only phonemes which occur in the examples are included in (21).

(21) Khalka Mongolian sonority classes (partial)

<table>
<thead>
<tr>
<th>phonemes</th>
</tr>
</thead>
<tbody>
<tr>
<td>vowel</td>
</tr>
<tr>
<td>resonant, nasal, glide</td>
</tr>
<tr>
<td>fricative</td>
</tr>
<tr>
<td>stop, affricate</td>
</tr>
</tbody>
</table>

The epenthetic vowel æ has not been included because its sonority is never relevant. It never appears unsyllabified.

Svantesson makes the following observations about KM syllable structure:

(22) a. Vowels must syllabify as nuclei; other phonemes cannot syllabify as nuclei.

b. There are no complex onsets.

 c. Codas must have strictly decreasing sonority.

Because of (22c), the three consonant nasal-fricative-stop coda in (20b) is maximal. Svantesson shows how the observed syllable patterns can be derived from the assumption of right to left syllable maximization under the constraints (22), inserting epenthetic æ as needed.

I will pursue a different line, adapting the ITB syllabification rule (18) to KM. I assume that (22a) is a statement of the Syllable Structure Constraints in KM. (22b) will follow from the weaker

---

2. This is called the “surface” form for want of a better term. It ignores variation in the surface form of the epenthetic vowel due to the local environment it appears in. The variation is irrelevant to the present inquiry.
assumption that *Complex Onset is imposed as avoidance constraint, as in ITB. (22c) will be a theorem. The main innovation that is needed is the introduction of a last resort syllable epenthesis subrule into the schema of elementary syllable building rules.

\[
\text{Left Epenthesis: } \times \rightarrow \sigma \times
\]

The targeted timing slot is integrated into the syllable structure as the coda of syllable whose nucleus is epenthetic.

Assuming the Sonority Hierarchy (21), KM syllabification is specified by:

(23) KM iterative syllabification:

a. SSC: Vowels must syllabify as nuclei; other phonemes cannot syllabify as nuclei.

b. Prominence = \[
\begin{bmatrix}
\text{Sonority} \\
\text{Right}
\end{bmatrix}
\]

c. \[
\mathcal{R} = \begin{bmatrix}
\text{Project} \\
\text{Adjoin Coda} \\
\text{Adjoin Onset} \\
\text{Left Epenthesis}
\end{bmatrix}; \text{ SSC } \& \begin{bmatrix}
*\text{Complex Onset} \\
*\text{No Onset}
\end{bmatrix}
\]

Aside from the different SSC, the differences with ITB are: 1) a last resort epenthesis rule;\(^3\) and 2) different secondary criteria entering into the preference relation.

The derivations of the syllable structure (20a,b,c) are straightforward.

(24) a. \[\text{bolžmr} \rightarrow \text{bolž.mar}\]

\[b < \overline{a} > 1 \times \tilde{z} < m < \overline{r} \]

b. \[\text{žimst} \rightarrow \text{žimst}\]

\[\tilde{z} < i > m > s > t \]

c. \[\text{tarwg} \rightarrow \text{tar.wəg}\]

\[t < \overline{a} > r < w < \overline{g} \]

The derivations of (20d,e,f) do not immediately produce the desired surface form.

---

\(^3\) It is worth observing that it would make very little difference to the predictions in ITB if the epenthesis rule were added to the schema of elementary syllabification rules. It would be called on only in examples like \text{təx.nəs}, producing \text{ət.xəz.nəs} instead. Ath-Sidar Rifian Berber, which we discuss later, produces such syllables under the analysis proposed. But they are eliminated by a later rule which deletes the epenthetic vowel and adjoins the consonant to the first syllable, producing the ITB result.
Onsetless syllables headed by an epenthetic vowel are produced in (25a,b). It would be possible without too much difficulty to modify the syllabification rule to change this behavior, but it will be clear when we turn to examine cyclic effects that this would be a mistake. Consequently, I take the cyclic syllabification rule to be what has been proposed and assume that there is a post-cyclic rule, perhaps very late in the derivation, which repairs onsetless syllables by stealing coda consonants when possible. Svantesson has a diphthong in place of the two vowels in hiatus in (25c). I tentatively assume that diphthong formation is a late rule as well, not part of iterative syllabification.4

The GDE account of KM syllabification is superior to the templatic account in an important way. The templatic account must stipulate that codas have decreasing sonority. The GDE account explains why this is so. It is a consequence of the way that the iteration proceeds, targeting local prominence maxima. The fact that in KM m syllabifies as an onset in bolzm, but n syllabifies as a coda in jortnc, has the same cause as the fact that in ITB n syllabifies as an onset in txznas and a nucleus in txznt.

3.1. Cyclic effects in syllabification

Svantesson gives a number of minimal pairs which demonstrate cyclic effects.

(26) a. xoc t la → xo.c@t.la
   ram VERB PAST
   ‘mounted (like a ram)’

b. xoc tl a → xo.c@.la
   bark TERM REFL
   ‘until its barking’

(27) a. zowl l → zo.wl.la
   advice PAST
   ‘advised’

b. zowl l o → zo.wl.los
   advice NOUN REFL
   ‘his advice’

(28) a. alt d ml → al.td.ml
   gold VERB ADJ
   ‘gilded’

b. ard cl l → ard.čr.la
   people VERB NOUN
   ‘democratization’

The sonority profiles of alt-d-ml and ard-cl-l are identical.

4. This is tentative because there is very little said about diphthongs in Svantesson’s paper, so there is little to go on.
The syllabification of the (a) examples above is what would be predicted if there were no cyclic syllabification and iterative syllabification applied to the concatenated form with morpheme boundaries irrelevant.

(29) a. \[ x \sigma \circ t l a \rightarrow x \sigma \circ t l a \rightarrow x \sigma \circ t l a \rightarrow x \sigma \circ t l a \rightarrow x \sigma \circ t l a \]

b. \[ z \sigma w l l o \rightarrow z \sigma w l l o \rightarrow z \sigma w l l o \rightarrow z \sigma w l l o \]

c. \[ a \sigma t d m l \rightarrow a \sigma t d m l \rightarrow a \sigma t d m l \rightarrow a \sigma t d m l \]

The (b) examples, however, demonstrate that the morpheme structure affects the syllabification of the inflected form.

If (30) is assumed, the desired results follow.

(30) Cyclic suffixation: When the suffix is concatenated, the syllable structure of the stem is reduced to core syllable structure and any \((\sigma)_o\) syllables that remain are deleted.

Reduction is illustrated below.

(31) a. \[ x \sigma \circ t l + a \rightarrow x \sigma \circ t l a \rightarrow x \sigma \circ t l a \]

b. \[ x \sigma \circ t l + a \rightarrow x \sigma \circ t l a \rightarrow x \sigma \circ t l a \]

Because the epenthetic syllable in (31a) has an onset, it does not delete when a suffix is concatenated. (31a) syllabifies as \(x\sigma c.t\sigma.la\) and (31b) as \(x\sigma.cat.la\).

The full derivations of (26) and (27) are given in (32). The suffixes are boxed and relevant prominence relations are shown where relevant. The crucial difference between the two derivations in each case is that in one an epenthetic syllable acquires an onset and is thereby protected from cyclic reduction. The steps in which an epenthetic syllable acquires an onset are indicated by †.
Svantesson illustrates the cyclic process with (33), which has 6 suffixes. All of the suffixed intermediate forms are possible surface words as well.

(33) a. uil ċl ul gč d iŋ x → uil.čol ‘to serve’
   action VERB CAUS AGENT PL GEN NOUN
   ‘belongings of the customers’

   b. morphology surface form meaning
   uil + ċl uil.čol ‘to serve’
   uil + ċl + ul uil.čul ‘to cause to serve’
   uil + ċl + ul + gč uilč.łuogč ‘customer’
   uil + ċl + ul + gč + d uilč.łuog.čad ‘customers’
   uil + ċl + ul + gč + d + iŋ uilč.łuog.čiŋ ‘of the customers’
   uil + ċl + ul + gč + d + iŋ + x uilč.łuog.čiŋx ‘belongings of the customers’

Below, the details of the derivation of (33a) are given, assuming (30).

(34) 1. u < l x l → u i l l → u i l l

2. u i l l > ċ < l → u i l l ċ l l → u i l l ċ l l
A potentially problematic aspect of the iterative syllabification rule that was proposed was the divergence between the cyclic forms and the surface forms, with onsetless syllables sometimes produced cyclically, but then repaired post-cyclically. The derivations (34) give strong evidence for the formulation of iterative syllabification that was proposed. If the succession of cyclic forms in the derivation above are compared with the succession of surface forms, the changes in the cyclic forms is regular and unremarkable, whereas the changes in the surface forms is obscure at best.

### Table 1: Morphology, Surface Structure, Cyclic Structure

<table>
<thead>
<tr>
<th>Cyclic Structure</th>
<th>Surface Structure</th>
<th>Morphology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. σ [u i l] &gt; ć &lt; l</td>
<td>uil.ćl</td>
<td>uil + ċl</td>
</tr>
<tr>
<td>2. σ [u i l] &gt; ć + ul</td>
<td>uil.ćl + ul</td>
<td>uil + ċl + ul</td>
</tr>
<tr>
<td>3. σ [u i l] + ul + gć</td>
<td>uil.ćl + ul + gć</td>
<td>uil + ċl + ul + gć</td>
</tr>
<tr>
<td>4. σ [u i l] + ul + gć + d</td>
<td>uil.ćl + ul + gć + d</td>
<td>uil + ċl + ul + gć + d + iŋ</td>
</tr>
<tr>
<td>5. σ [u i l] + ul + gć + d + iŋ</td>
<td>uil.ćl + ul + gć + d + iŋ</td>
<td>uil + ċl + ul + gć + d + iŋ + x</td>
</tr>
<tr>
<td>6. σ [u i l] + ul + gć + d + iŋ + x</td>
<td>uil.ćl + ul + gć + d + iŋ + x</td>
<td>uil + ċl + ul + gć + d + iŋ + x</td>
</tr>
</tbody>
</table>

The only point that requires comment in the sequence of cyclic forms is that an epenthetic vowel deletes when it loses its coda to a V-initial suffix and only (⟨ə⟩)ₜ remains.

If we had supposed that the cyclic derivation operates on something close to the surface forms, it is hard to see how the surface forms would be produced. Most obviously, the questions below are raised.

### Table 2: Input, Surface Form, Why Not?

<table>
<thead>
<tr>
<th>Input</th>
<th>Surface Form</th>
<th>Why Not?</th>
</tr>
</thead>
<tbody>
<tr>
<td>uil.ćl + ul</td>
<td>uil.ćl.lul</td>
<td>uil.ćl.lul</td>
</tr>
<tr>
<td>uil.ćl + ul + gć</td>
<td>uil.ćl.lul</td>
<td>uil.ćl.lul</td>
</tr>
<tr>
<td>uil.ćl + ul + gć + d</td>
<td>uil.ćl.lul</td>
<td>uil.ćl.lul</td>
</tr>
<tr>
<td>uil.ćl + ul + gć + d + iŋ</td>
<td>uil.ćl.lul</td>
<td>uil.ćl.lul</td>
</tr>
<tr>
<td>uil.ćl + ul + gć + d + iŋ + x</td>
<td>uil.ćl.lul</td>
<td>uil.ćl.lul</td>
</tr>
</tbody>
</table>
In spite of its complexity, (34) does not furnish evidence that syllabification must be cyclic. If
the forms are syllabified ignoring morpheme boundaries, the desired results are obtained. But we
know from (32) that there must be cyclic syllabification. What (34) does is furnish considerable
evidence for the approach to cyclic syllabification (30) that was proposed. The derivation produces
all of the intermediate forms, so there are multiple tests.

4. Ath-Sidar Rifian Berber

This section is based on Dell and Tangi (1993). All of the data and empirical generalizations are
theirs. The theoretical framework is so different that it is hard to point to specific aspects of their
analysis that have been incorporated, but there is considerable influence throughout.

Ath-Sidar Rifian Berber (ASR) syllabification of input phoneme sequences which do not
involve r is for the most part quite familiar. Some examples follow in (37). Epenthetic @ is used
to break up consonant sequences which could not otherwise be syllabified. Morpheme boundaries
are shown in the input. # denotes a clitic boundary and the other morpheme boundaries are denoted
by -. The latter do not affect the output, but clitic boundaries can (see Section 4.2).

(37) underlying surface
   a. nudm-n nud.m@n ‘they dozed’
   b. 3q@-n 5q@.3m@n ‘I pulled’
   c. nqs nq@s ‘decrease’

The syllabification of most forms containing r is more complex.

(38) underlying surface
   a. frin frin ‘sort (neg stem)’
   b. frn fan ‘sort (perf stem)’
   c. !r@a la .@a ‘gratefully accept’
   d. !r Sf l a . 5f ‘make noise’
   e. fsr fsa ‘hang up washing’

One might conclude on the basis of data like these that there is an r→a rule that applies when r
is not an onset, as it is in (38a). However, if r is the first timing slot of a geminate, both a and r
appear.

(39) underlying surface
   a. frn fr@.r@n ‘sort (imperf. stem)’
   b. rr Sf arr . 5f ‘uproar’

This suggests that there are two rules, a-insertion and r-deletion, with a-insertion applying in both
(38) and (39), but with r-deletion blocked in (39) by geminate inalterability. We pursue this line.5

As a starting point, assume that the syllabification rule is identical to the KM rule, with
the differences in syllabification due to differences in the Sonority Hierarchy and the Syllable
Structure Conditions.

5. Dell and Tangi’s preliminary analysis combines a insertion and r deletion, but they abandon it in favor of an
‘r-vocalization’ analysis, for reasons that are not entirely clear to me.
Iterative syllabification

a. Sonority Hierarchy (restricted to phonemes which appear in the examples)

phonemes
vowels a, i, u
r
other q, d, ð, t, h, x, f, s, z, ñ, m, ñ, y, w

b. Prominence = Sonority
Right

c. SSC: Vowels must be nuclei. Other phonemes cannot be nuclear.

d. \( \mathcal{R} = [ \text{Project Syllable} \ \ \ \text{Adjoin Onset} \ \ \ \ \ \text{Adjoin Coda} \ \ \ \ \ \text{Left Epenthesis} ] \); SSC & *Complex Onset *

*No Onset *

As in ITB and ASR, the growth loci are targeted simultaneously.

This is a starting point. A few amendments will be needed as we proceed. I assume as well that there are rules A-INS and R-DEL which apply (in that order) at some point after iterative syllabification. A-INS replaces epenthetic \( \sigma \) with \( a \) if the coda is \( r \). R-DEL deletes coda \( r \) if the nucleus is \( a \).

The derivations of (37a,b) and (38b,c) are straightforward. The steps after iterative syllabification below (applications of A-INS and R-DEL) are demarcated by a horizontal line.

(41) a. \( 5bðñ \rightarrow 5ð ñ \) b. \( nudm \rightarrow nu.ðm \) c. \( fm \rightarrow fan \) d. \( !rða \rightarrow !a.ða \)

\[
\begin{array}{cccc}
5 < b < ð < ñ & n < u > d < m < n & f < r > n & !f > ð < a \\
3 < b < ð & 2 & n & 2 & n & f \sigma & r \sigma & 2 & \sigma & \sigma & \sigma & \sigma \\
3 < b & ð & 2 & n & 2 & n & f \sigma & r \sigma & a \sigma & \sigma & \sigma & \sigma \\
3 \sigma & 2 & b & ð & 2 & n & 2 & n & f \sigma & r \sigma & a \sigma & \sigma & \sigma & \sigma \\
5 \sigma & 2 & b & ð & 2 & n & \sigma & \sigma & \sigma & \sigma \\
\end{array}
\]

The other examples in (37) and (38) do not immediately produce the desired outcome.
(42) a. \( nqs \rightarrow nq\sigma s \)  
\[
\begin{array}{c}
\sigma \\
\sigma
\end{array}
\]
\( n < q < s \) 
\[
\begin{array}{c}
f \\
\sigma
\end{array}
\]
\( f < r < i > n \) 
\[
\begin{array}{c}
f \\
\sigma
\end{array}
\]
\( f < s < r \) 
\[
\begin{array}{c}
s \\
\sigma
\end{array}
\]
\( s < f \) 

b. \( frin \rightarrow fr\alpha n \) 
\[
\begin{array}{c}
f \\
\sigma
\end{array}
\]
\( f < r < i > n \) 
\[
\begin{array}{c}
f \\
\sigma
\end{array}
\]
\( f < s < r \)
\[
\begin{array}{c}
s \\
\sigma
\end{array}
\]
\( s < f \) 

Just as in KM, iterative syllabification sometimes produces noninitial onsetless syllables, as in (42d). As in KM, I assume that they are eliminated by a late rule (ONS) which steals a coda from the previous syllabic. Unlike KM, initial onsetless syllables with an epenthetic nucleus are common. They cannot be repaired by coda theft. In ASR, there are two operations which are used to repair such syllables. Both delete the epenthetic vowel, but differ in the disposition of the orphaned consonant. If the syllable is \( (\sigma y)_\sigma \) or \( (\sigma w)_\sigma \), with the coda a glide, the epenthetic vowel is deleted and the glide vocalizes. Otherwise, the epenthetic vowel deletes and the stranded consonant adjoins to the adjacent syllable. This applies to (42a,b,c). Example (44a) below exemplifies glide vocalization. Both varieties of repair will be called E-DEL.

4.1. Geminates

As in ITB, what becomes the second slot of a geminate is bare at the point of syllabification and can be syllabified as an onset or coda, but not a nucleus. This is illustrated in (43).
4.2. Cyclic effects

If an r final stem combines with a vowel initial clitic, and syllabification is assumed to follow concatenation, the syllabification rules proposed do not make the correct predictions.

(44) 

<table>
<thead>
<tr>
<th>input</th>
<th>predicted</th>
<th>actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>n-final stem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a. y-ðhn # i t</td>
<td>i.ðōh.ni t</td>
<td>i.ðōh.ni t ‘he rubbed her’</td>
</tr>
<tr>
<td>r-final stem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. y-!ndr # i t</td>
<td>!y.ənd.r i t</td>
<td>!yən.də.r i t ‘he threw (perf.) it (f.)’</td>
</tr>
<tr>
<td>c. !i-uzr # a</td>
<td>!i.əz.r a</td>
<td>!iəz.rə ‘river’</td>
</tr>
<tr>
<td>d. 'a-zwr # a</td>
<td>‘a.zəw.r a</td>
<td>‘aəz.wərə ‘blood vessel’</td>
</tr>
</tbody>
</table>

If the stem final phoneme is not r, (44a) above for example, there is no evidence of cyclic syllabification. The correct prediction is made if it is assumed that no syllabification takes place until after the clitic is concatenated with the stem.

(45) 

\[ y \ddot{d} h n + i t \rightarrow y<\ddot{d}<h<n<i>t \rightarrow y<\ddot{d}<h\hat{n}1t \rightarrow y<\ddot{d}\hat{h}n1t \]

In order to distinguish the r-final cases from the others, D&T proposed that clitics attach to a syllabified stem but that when the clitic is concatenated with the stem, the syllable structure of the stem is reduced, as in Khalka Mongolian (where the process applies to all affixes, not just clitics). The reduction is even more aggressive than it is in KM, reducing the syllable structure to core syllables and deleting all the epenthetic vowels (in KM, only epenthetic vowels in onsetless syllables are deleted). Crucially, A-INS is cyclic, so that the position of epenthetic vowels that have been replaced by a escape deletion when the clitic is concatenated. This is somewhat similar to cyclic syllabification in Khalka Mongolian. In KM, epenthetic syllables which have acquired an onset cyclically are protected from deletion when a suffix is concatenated.

(46) 

1. \[ y<\ddot{n}<d<r \rightarrow y<\ddot{n}<d2r \rightarrow \ldots \rightarrow y\ddot{a}n\ddot{d}2r \rightarrow y\ddot{a}n\ddot{d}r \]

2. \[ y<\ddot{n}<d\ddot{a}r<i>t \rightarrow y\ddot{a}n\ddot{d}ar1t \rightarrow y\ddot{a}n\ddot{d}ar1t \] (stem reduction)
1. \( y < \delta < h < n \rightarrow y \delta h \sigma n \rightarrow \ldots \rightarrow y \sigma h \sigma n \)

\[
\begin{array}{c}
y \sigma h \sigma n \rightarrow y \sigma h \sigma n \rightarrow y \sigma h \sigma n \rightarrow y \sigma h \sigma n \\
2. (stem reduction)
\end{array}
\]

2. \( y < \delta < h < n < \| t \rightarrow \ldots \rightarrow y \sigma h \sigma n \sigma \| t \) (as in (45) above)

Note that the cyclic effects in (44b,c,d) are further evidence that \( a \)-insertion and \( r \)-deletion are separate operations, since \( a \)-insertion must apply without \( r \)-deletion.

### 4.3. Word final glide vocalization

We have already encountered one of the effects of \( y \) and \( w \) being treated generally like consonants, but vocalizing in particular environments. \( ay \rightarrow i \) and \( ow \rightarrow u \) under certain circumstances as a way to eliminate a \( \sigma \)-initial syllable.

A more striking effect of glide vocalization is that word final glides are generally treated by the syllabification rules as if they have intermediate sonority (i.e. the sonority of \( r \)) and are optionally nuclear. This is illustrated by the contrasts between (48) and (49).

\begin{align*}
(48) & \text{underlying} & \text{surface} \\
& a. \ fsy & fsi & \text{‘untie (imperf. 2sg)’} \\
& \quad \quad \text{fsy-\( t \)} & f\sigma s.y\sigma t & \text{‘untie (imperf. 2pl)’} \\
& b. \ hry & hri & \text{‘grind (imperf. 2sg)’} \\
& \quad \quad \text{hry-\( t \)} & h\sigma a.y\sigma t & \text{‘grind (imperf. 2pl)’} \\
& c. \ tt-arw & t\sigma t.a.r\sigma u & \text{‘give birth (imperf. stem)’} \\
& tt-arw-\text{\( w \)} & t\sigma t.a.w\sigma & \text{‘give birth (imperf. 1sg)’}
\end{align*}

If the sonority of the final glide were not exceptionally raised in (48b), the result would be different. Iterative syllabification would initially target \( r \) and an epenthetic vowel inserted.

The derivation of (49) is interesting because it gives evidence that \( r \)-deletion occurs independently of \( a \)-insertion. (49.1) and (49.2) are the two cycles of cyclic syllabification and (49.2) is the post-cyclic derivation.

\begin{align*}
(49) & \text{tt-arw} \rightarrow \text{tt.a.w} \\
1. \ a > r > w < \| B \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \\
2. \ b > x < a > r > w < \| B \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \\
3. \ \text{ONS} \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t \rightarrow IS \sigma r \rightarrow IS \sigma t
\end{align*}

Crucially, since \( r \) appears directly after \( a \) in the underlying representation, it becomes a coda of a syllable whose nucleus is not epenthetic so it does not trigger \( a \)-insertion. Later, it deletes, like all instances of \( r \) as an \( a \)-coda.
The account we gave of the examples (38) involved two separate rules; one inserts \( a \) and the other deletes \( r \). We now have evidence that both of these rules are independently needed in the phonology. Earlier, we gave examples where \( a \) was inserted, but \( r \) did not delete because after insertion other rules altered the environment in which \( r \) appeared. Example (49) shows that a rule that deletes \( r \) when it appears as an \( a \)-coda is needed independently of \( a \)-insertion. Taken together, this is good evidence for an explanation of the (38) examples in terms of two separate rules.

5. Conclusion

The main result of this study is the demonstration that Dell and Elmedlaoui’s algorithm for core syllabification in ITB can be extended to provide analyses of languages in which epenthesis plays a major role in syllabification. The appeal of the influential templatic theory of syllabification proposed by Ito (1986, 1989) has been that it integrates epenthesis into the syllabification process itself. This insight has been carried over into the theory of sonority driven syllabification developed in this study, but in a different form. It is proposed that there are complex syllabification rules which simultaneously build syllable structure and epenthesize nuclei. For example

\[
\begin{array}{c}
\times \rightarrow \sigma \\
\text{C} \quad \text{C}
\end{array}
\]

This minimal combination of epenthesis with syllable building is all that is required. Languages have the choice of epenthetic vowels and epenthesis to the left rather than to the right. Although in fact very simple, I call the rule “complex” because it does two different things. It builds syllable structure and epenthesizes a vowel. Of course, if one thinks of epenthesis as part of syllable building, as Ito convincingly argued that one should do, it is not clear that they should be thought of as two different things. Be that as it may, it is crucial that the theory admits rules of this kind.

It is my view that sonority-driven syllabification based on local interactions can supplant templatic theory, explaining all the phenomena that Ito explained as well as phenomena that cannot be explained in templatic terms. This remains to be seen. If so, syllable templates are epiphenomenal. This result would be welcome, because it would show that the language faculty has fewer computational devices available to it than if templates turn out to be necessary.
References